MAX TNT Administration Guide

Ascend Communications, Inc.
Part Number: 7820-0548-002
For software version 7.0.0

Ascend Customer Service

You can request assistance or additional information by telephone, email, fax, or modem, or over the Internet.

Obtaining Technical Assistance

If you need technical assistance, first gather the information that Ascend Customer Service will need for diagnosing your problem. Then select the most convenient method of contacting Ascend Customer Service.

Information you will need

Before contacting Ascend Customer Service, gather the following information:

- Product name and model
- Software and hardware options
- Software version
- Type of computer you are using
- Description of the problem

How to contact Ascend Customer Service

After you gather the necessary information, contact Ascend in one of the following ways:

Telephone in the United States	800-ASCEND-4 (800-272-3634)
Telephone outside the United States	510-769-8027 (800-697-4772)
Austria/Germany/Switzerland	(+33) 492 96 5672
Benelux	(+33) 492 96 5674
France	(+33) 492 96 5673
Italy	(+33) 492 96 5676
Japan	(+81) 3 5325 7397
Middle East/Africa	(+33) 492 96 5679
Scandinavia	(+33) 492 96 5677
Spain/Portugal	(+33) 492 96 5675
UK	(+33) 492 96 5671
Email	support@ascend.com
Email (outside US)	EMEAsupport@ascend.com
Facsimile (FAX)	510-814-2312
Customer Support BBS by modem	510-814-2302

You can also contact the Ascend main office by dialing 510-769-6001, or you can write to Ascend at the following address:

Ascend Communications
1701 Harbor Bay Parkway
Alameda, CA 94502

Need information about new features and products?

Ascend is committed to constant product improvement. You can find out about new features and other improvements as follows:

- For the latest information about the Ascend product line, visit our site on the World Wide Web:

  `http://www.ascend.com`

- For software upgrades, release notes, and addenda to this manual, visit our FTP site:

  `ftp.ascend.com`

Important safety instructions

The following safety instructions apply to the MAX TNT:

1 Read and follow all warning notices and instructions marked on the product or included in the manual.

2 The maximum recommended ambient temperature for MAX TNT models is 104° Fahrenheit (40° Celsius). Care should be given to allow sufficient air circulation or space between units when the MAX TNT is installed in a closed or multi-unit rack assembly, because the operating ambient temperature of the rack environment might be greater than room ambient.

3 Slots and openings in the cabinet are provided for ventilation. To ensure reliable operation of the product and to protect it from overheating, these slots and openings must not be blocked or covered.

4 Installation of the MAX TNT in a rack without sufficient air flow can be unsafe.

5 If installed in a rack, the rack should safely support the combined weight of all equipment it supports. A fully loaded redundant-power MAX TNT weighs 130 lbs (58.97 kg).

6 The connections and equipment that supply power to the MAX TNT should be capable of operating safely with the maximum power requirements of the MAX TNT. In the event of a power overload, the supply circuits and supply wiring should not become hazardous. The input rating of the MAX TNT is printed on its nameplate.

7 Models with AC power inputs are intended to be used with a three-wire grounding type plug - a plug which has a grounding pin. This is a safety feature. Equipment grounding is vital to ensure safe operation. Do not defeat the purpose of the grounding type plug by modifying the plug or using an adapter.

8 Prior to installation, use an outlet tester or a voltmeter to check the AC receptacle for the presence of earth ground. If the receptacle is not properly grounded, the installation must not continue until a qualified electrician has corrected the problem. Similarly, in the case of DC input power, check the DC ground (s).

9 If a three-wire grounding type power source is not available, consult a qualified electrician to determine another method of grounding the equipment.

10 Models with DC power inputs must be connected to an earth ground through the terminal block Earth/Chassis Ground connectors. This is a safety feature. Equipment grounding is vital to ensure safe operation.

11 Prior to installing wires to the MAX TNT unit's DC power terminal block, verify that these wires are not connected to any power source. Installing live wires (that is, wires connected to a power source) is hazardous.

12 Connect the equipment to a 48 VDC supply source that is electrically isolated from the AC source. The 48VDC source should be reliably connect to earth.

13 Install only in restricted access areas in accordance with Articles 110-16, 110-17, and 110-18 of the National Electrical Code, ANSI/NFPA 70.

14 Do not allow anything to rest on the power cord and do not locate the product where persons will walk on the power cord.

15 Do not attempt to service this product yourself, as opening or removing covers may expose you to dangerous high voltage points or other risks. Refer all servicing to qualified service personnel.

16 General purpose cables are provided with this product. Special cables, which may be required by the regulatory inspection authority for the installation site, are the responsibility of the customer.

17 When installed in the final configuration, the product must comply with the applicable Safety Standards and regulatory requirements of the country in which it is installed. If necessary, consult with the appropriate regulatory agencies and inspection authorities to ensure compliance.

Warning: A rare phenomenon can create a voltage potential between the earth grounds of two or more buildings. If products installed in separate buildings are **interconnected**, the voltage potential may cause a hazardous condition. Consult a qualified electrical consultant to determine whether or not this phenomenon exists and, if necessary, implement corrective action prior to interconnecting the products.

Warning: To reduce the risk of fire, communication cable conductors must be 26 AWG or larger.

Warning: Afin de reduire les risques d'incendie, les fils conducteurs du cable de communication doivent etre d'un calibre minimum de 26 AWG (American Wire Gauge), cest-a-dire d'un minimum de 0,404 mm.

In addition, if the equipment is to be used with telecommunications circuits, take the following precautions:

- Never install telephone wiring during a lightning storm.
- Never install telephone jacks in wet locations unless the jack is specifically designed for wet locations.
- Never touch uninsulated telephone wires or terminals unless the telephone line has been disconnected at the network interface.
- Use caution when installing or modifying telephone lines.

- Avoid using equipment connected to telephone lines (other than a cordless telephone) during an electrical storm. There is a remote risk of electric shock from lightning.
- Do not use a telephone or other equipment connected to telephone lines to report a gas leak in the vicinity of the leak.

Contents

Figures

Tables

Introduction

1

This introduction covers the following topics:

What is in this guide

This guide describes how to manage the MAX TNT and troubleshoot its operations. It assumes that you have set up the MAX TNT system as described in the *MAX TNT Glossary* and configured it for network connectivity as described in the *MAX TNT Network Configuration Guide*.

Each chapter in the guide focuses on a particular aspect of MAX TNT administration and operations. The chapters describe tools for system management, network management, and SNMP management.

Although some of the sections in this manual deal with security issues, the *MAX TNT Network Configuration Guide* provides a more comprehensive approach to such topics as securing the unit, using firewalls, and understanding the more complex authentication procedures (such as the use of dynamic passwords).

To perform many of the tasks in this manual, you must have administrative permission on the MAX TNT. For instructions on logging into the MAX TNT with administrative permissions, see “Logging into the MAX TNT” on page 2-2.

What you should know

While this guide attempts to provide a conceptual framework that will sufficiently enable an administrator who is not an expert in a particular network technology to operate and troubleshoot the unit, it does not start from the beginning with any network management topic. Following are the general areas in which it is helpful to have some existing knowledge when working with the related capability in the MAX TNT:

- Line configuration and testing
- Dial-in connections such as PPP or clear TCP
- Connection negotiation and authentication

- Connection-cost management and accounting
- ISDN
- Modems
- Frame Relay
- IP routing
- OSPF routing (if applicable)
- Multicast (if applicable)
- Network security

Related publications

Additional information is available in the other guides in the MAX TNT documentation set. If you need more background information than these guides provide, many external references are readily available on the Web or in technical bookstores. You'll find a partial list of such references below.

MAX TNT documentation set

The MAX TNT documentation set consists of the following manuals:

- *The Ascend Command-Line Interface*. Shows how to use the MAX TNT command-line interface effectively.
- *MAX TNT Administration Guide* (this manual). Contains troubleshooting and administrative information.
- *MAX TNT Hardware Installation Guide*. Shows how to install the MAX TNT hardware and configure its shelf controller and slot cards for a variety of supported uses. Describes how calls are routed through the system. Includes the MAX TNT technical specifications and some administrative information.
- *MAX TNT Glossary*. Defines networking terms and concepts used in the MAX TNT documentation.
- *MAX TNT Network Configuration Guide*. Describes how to use the command-line interface to configure WAN connections and other related features.
- *MAX TNT RADIUS Configuration Guide*. Describes how to use install and configure RADIUS.
- *MAX TNT Reference Guide*. An alphabetic reference to all MAX TNT profiles, parameters, and commands.

Related RFCs

RFCs are available on the Web at http://ds.internic.net.

Information about PPP connections

For information about PPP connections and authentication, you might want to download one or more of the following:

- RFC 1332: *The PPP Internet Protocol Control Protocol (IPCP)*

- RFC 1618: *PPP over ISDN*
- RFC 1638: *PPP Bridging Control Protocol (BCP)*
- RFC 1661: *The Point-to-Point Protocol (PPP)*
- RFC 1662: *PPP in HDLC-like Framing*
- RFC 1877: *PPP Internet Protocol Control Protocol Extensions for Name Server Addresses*
- RFC 1934: *Ascend's Multilink Protocol Plus (MP+)*
- RFC 1962: *The PPP Compression Control Protocol (CCP)*
- RFC 1974: *PPP Stac LZS Compression Protocol*
- RFC 1989: *PPP Link Quality Monitoring*
- RFC 1990: *The PPP Multilink Protocol (MP)*
- RFC 1994: *PPP Challenge Handshake Authentication Protocol (CHAP)*
- RFC 2125: *The PPP Bandwidth Allocation Control Protocol (BACP)*
- RFC 2153: *PPP Vendor Extensions*

Information about IP routers

RFCs that describe the operation of IP routers include:

- RFC 1256: *ICMP Router Discovery Messages*
- RFC 1393: *Traceroute Using an IP Option*
- RFC 1433: *Directed ARP*
- RFC 1519: *Classless Inter-Domain Routing (CIDR): An Address Assignment and Aggregation Strategy*
- RFC 1582: *Extensions to RIP to Support Demand Circuits*
- RFC 1787: *Routing in a Multi-provider Internet*
- RFC 1812: *Requirements for IP Version 4 Routers*
- RFC 2002: *IP Mobility Support*
- RFC 2030: *Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI*

Information about OSPF routing

For information about OSPF routing, see:

- RFC 1245: *OSPF protocol analysis*
- RFC 1246: *Experience with the OSPF protocol*
- RFC 1583: *OSPF Version 2*
- RFC 1586: *Guidelines for Running OSPF Over Frame Relay Networks*
- RFC 1587: *The OSPF NSSA Option*
- RFC 1850: *OSPF Version 2 Management Information Base*

Information about multicast

For information about multicast, see:

- RFC 1458: *Requirements for Multicast Protocols*
- RFC 1584: *Multicast Extensions to OSPF*
- RFC 1949: *Scalable Multicast Key Distribution*

Information about firewalls and packet filtering

RFCs that describe firewalls and packet filters include:

- RFC 1579: *Firewall-Friendly FTP*
- RFC 1858: *Security Considerations for IP Fragment Filtering*

Information about general network security

RFCs pertinent to network security include:

- RFC 1244: *Site Security Handbook*
- RFC 1281: *Guidelines for the Secure Operation of the Internet*
- RFC 1636: *Report of IAB Workshop on Security in the Internet Architecture*
- RFC 1704: *On Internet Authentication*

Information about external authentication

For information about RADIUS and TACACS authentication, see:

- RFC 2138: *Remote Authentication Dial In User Service (RADIUS)*
- RFC 1492: *An Access Control Protocol, Sometimes Called TACACS*

ITU-T recommendations

ITU-T recommendations (formerly CCITT) are available commercially. You can order them at http://www.itu.ch/publications/.

Related books

The following books are available in technical bookstores.

- *Routing in the Internet*, by Christian Huitema. Prentice Hall PTR, 1995. Recommended for information about IP, OSPF, CIDR, IP multicast, and mobile IP.
- *SNMP, SNMPV2 and RMON: Practical Network Management*, by William Stallings. Addison-Wesley, 1996. Recommended for network management information.
- *Enterprise Networking: Fractional T1 to Sonet Frame Relay to Bisdn*, by Daniel Minoli. Artech House, 1993. Recommended as a WAN reference.
- *TCP/IP Illustrated*, volumes 1&2, by W. Richard Stevens. Addison-Wesley, 1994.

Documentation conventions

Table 1-1 shows the documentation conventions used in this guide.

Table 1-1. Documentation conventions

Convention	Meaning
`Monospace text`	Represents text that appears on your computer's screen, or that could appear on your computer's screen.
`Boldface monospace text`	Represents characters that you enter exactly as shown (unless the characters are also in ***`italics`***—see *Italics*, below). If you could enter the characters, but are not specifically instructed to, they do not appear in boldface.
Italics	Represent variable information. Do not enter the words themselves in the command. Enter the information they represent. In ordinary text, italics are used for titles of publications, for some terms that would otherwise be in quotation marks, and to show emphasis.
[ ]	Square brackets indicate an optional argument you might add to a command. To include such an argument, type only the information inside the brackets. Do not type the brackets unless they appear in bold type.
|	Separates command choices that are mutually exclusive.
>	Points to the next level in the path to a parameter. The parameter that follows the angle bracket is one of the options that appears when you select the parameter that precedes the angle bracket.
Key1-Key2	Represents a combination keystroke. To enter a combination keystroke, press the first key and hold it down while you press one or more other keys. Release all the keys at the same time. (For example, Ctrl-H means hold down the Control key and press the H key.)
Press Enter	Means press the Enter, or Return, key or its equivalent on your computer.
Note:	Introduces important additional information.
 Caution:	Warns that a failure to follow the recommended procedure could result in loss of data or damage to equipment.
 **Warning:**	Warns that a failure to take appropriate safety precautions could result in physical injury.

MAX TNT System Administration

2

This chapter covers the following topics:

This chapter explains how to perform common system administration tasks on the MAX TNT. It focuses on tasks you can perform on the system as a whole, such as resetting the unit, setting the time and date, configuring logging, and backing up and restoring a configuration. For information about managing the MAX TNT slot cards, see Chapter 3, "Administering MAX TNT Slot Cards."

Logging into the MAX TNT

To administer the system, you can log in from a PC connected to the MAX TNT unit's serial port, or from a workstation that has Telnet access to the system. When you log in, you are prompted for a user name:

```
User:
```

To log in with administrative (superuser) privileges, enter the default password (`Ascend`) assigned to the MAX TNT Admin login at the factory:

```
User: admin
Password: Ascend
```

The name specified in the Admin User profile appears as your system prompt. For example:

```
admin>
```

If you are already connected to the MAX TNT as a different user, use the Auth command to log in as the administrator:

```
admin> auth admin
Password:
```

Note: Because the Admin login has superuser privileges, you should change the default password immediately. Be sure to write down the password you assign and store it in a safe place.

Following is an example of changing the password for the Admin login:

```
admin> read user admin
USER/admin read

admin> set password = top-secret

admin> write
USER/admin written
```

All subsequent administrator logins will be required to supply the new password. (For more information about configuring User profiles, see Chapter 7, "Creating User Profiles.")

Securing the serial port

By default, when users connect to the serial port on the shelf controller, they are logged in with the Admin User profile. To secure the serial port with a username and password, proceed as follows:

1 Read the Serial profile:

```
admin>read serial { 1 17 2}
```

2 Set the User-Profile to null:

```
admin>set user =
```

3 Set Auto-Logout to Yes:

```
admin>set auto-logout = yes
```

This automatically logs out the current User profile if DTR is lost on the serial port.

4 Write the profile:

```
admin>write
```

Now users connecting to the serial port must supply a valid username and password for access to the MAX TNT.

Specifying a management-only Ethernet interface

You can specify that one of the MAX TNT Ethernet interfaces is for management only. The management-only interface can be the shelf-controller port or a port on an installed Ethernet card. Following is the relevant parameter, which is shown with its default setting:

```
[in IP-INTERFACE/{ { any-shelf any-slot 0 } 0 }]
management-only-interface = no
```

Setting Management-Only-Interface to Yes means that incoming traffic on the interface terminates in the system itself. It is not forwarded on any other interface. In addition, only traffic generated by the system is forwarded on the management-only interface. Traffic generated externally is dropped on the interface.

To configure a management interface, proceed as in the following example:

```
admin> read ip-int {{ 1 12 1 } 0}
IP-INTERFACE/{ { shelf-1 slot-12 1 } 0 } read

admin> set management-only = yes

admin> write
IP-INTERFACE/{ { shelf-1 slot-12 1 } 0 } written
```

The IfMgr –d command displays a Management Only field to reflect the port's status.

Overview of MAX TNT commands

Each card in the MAX TNT has its own set of commands. The commands on the shelf controller typically affect the operation of the entire system. The commands on particular cards, such as the T1 or Ethernet cards, affect only the cards themselves. This section explains the commands available on the shelf controller.

For information about commands available on the cards, see Chapter 3, "Administering MAX TNT Slot Cards," or the *MAX TNT Reference Guide*. For information on debug commands, see Chapter 5, "Using the MAX TNT Debug Commands."

Command permission-levels

Commands are organized by permission levels, as described in Table 2-1. A user gains access to a particular command by logging in to the MAX TNT by means of a user profile that specifies the required permission level. (To create a User profile, see Chapter 7, "Creating User Profiles.") By default, the Admin profile specifies permission to execute all commands.

Table 2-1. Permission levels

Permission level	Description
Code	Allows you to format and manage the PCMCIA cards that store the system software.
Debug	Specialized commands used to troubleshoot the cards. Under most circumstances, these commands are not required for correct operation of the MAX TNT, and in some circumstances might produce undesirable results. (For information about the debug commands, see Chapter 5, "Using the MAX TNT Debug Commands."
Diagnostic	Commands used to monitor the MAX TNT and its cards.
System	Commands that allow you to manage and configure the MAX TNT.
Term-Serv	Accesses the MAX TNT terminal server.
Update	Commands that allow you to update the system configuration.
User	Simple commands available to all users that allow log in.

Commands overview

Table 2-2 briefly describes the MAX TNT commands available on the shelf-controller. Many of the commands are used in later sections of this manual to perform certain system administration tasks. For complete details of each command, see the *MAX TNT Reference Guide*.

Table 2-2. MAX TNT system administration commands

Command Name	Permission Level	Effect
?	User	Displays a list of commands.
Arptable	System	Displays or modify the MAX TNT Address Resolution Protocol (ARP) table.
Auth	User	Selects a new User profile.
BRIchannels	System	Displays IDSL line information.
CADSLlines	System	Displays ADSL-CAP line information.

Table 2-2. MAX TNT system administration commands (continued)

Command Name	Permission Level	Effect
Callroute	Diagnostic	Displays the call-routing database.
Clear	User	Clears the terminal session screen and places the system prompt at the top row of the VT100 window.
Clock-Source	Diagnostic	Displays clock-source statistics.
Clr-History	System	Clears the fatal-error history log.
Connection	System	Displays the connection-status window.
DADSLlines	System	Displays ADSL-DMT line information.
Date	Update	Sets the system date.
Debug	Diagnostic	Enables or disable diagnostic output.
Delete	Update	Permanently deletes a profile from local storage.
Device	Diagnostic	Brings a device up or down.
Dir	System	Lists profiles and profile types.
Dircode	System	Shows contents of PCMCIA card code.
Dnstab	System	Displays DNS table entries.
DS3ATMlines	System	Displays DS3 ATM line information.
Ether-Display	Diagnostic	Displays contents of received Ethernet packets.
Help	User	Displays help about a particular command.
Fatal-History	System	Lists fatal-error history log.
Format	Code	Prepares a flash card for use.
Fsck	Code	Verifies the filesystem on a PCMCIA flash card. If errors are detected, they are reported. No errors are fixed.
Get	System	Displays fields in a profile.
HDLC	System	Displays HDLC-channel information.
If-Admin	Diagnostic	Administer an interface.
IGMP	System	Displays IGMP multicast statistics.

Table 2-2. MAX TNT system administration commands (continued)

Command Name	Permission Level	Effect
IP-pools	System	Displays the status of the IP address pools configured in the IP-Global profile.
Ipcache	System	Displays IP route caches.
IProute	System	Enables you to manually add or delete IP routes. Routing table changes made by using this command are not remembered across system resets.
Line	System	Displays the line status window.
List	System	Lists fields in working profile.
Load	Update	Uploads code or saved configuration to flash.
Log	System	Invokes/controls the event log window.
Modem	System	Displays modem information.
Netstat	System	Displays routing or interface tables.
New	System	Creates a new profile.
NSlookup	Diagnostic	Resolves the IP address of a specified host name by performing a DNS lookup.
Nvram	Update	Clears configuration and reboot system
OAMloop	Diagnostic	Sends ATM Operation-And-Maintenance (OAM) loop-back cells on an ATM interface.
Open	Diagnostic	Starts session with slot card.
OSPF	System	Displays information related to OSPF routing, including Link-State Advertisements (LSAs), border routers' routing tables, and the OSPF areas, interfaces, statistics, and routing table.
Ping	Diagnostic	Sends ICMP echo_request packets to the specified host as a way to verify that the host is up and the transmission path to the host is open.
Power	System	Displays power supply statistics.
Quiesce	System	Temporarily disables a modem or DS0 channel.
Read	System	Makes the specified profile the working profile.

Table 2-2. MAX TNT system administration commands (continued)

Command Name	Permission Level	Effect
Refresh	System	Refreshes the remote configuration.
Reset	Update	Reboots the system.
Save	Update	Saves profile for future restore.
Screen	System	Changes the status window display size for the current session.
SDSLlines	System	Displays SDSL line information.
Set	System	Sets a parameter's value.
Show	System	Shows shelves, slots, or items.
Slot	Diagnostic	Administers a slot card.
Status	System	Displays system status or hide status window.
SWANlines	System	Displays serial WAN line information.
T1channels	System	Displays T1 channel information.
Telnet	Diagnostic	Opens a Telnet session to another host.
Terminal-Server	Termserv	Enters terminal-server mode.
Traceroute	Diagnostic	Traces the route an IP packet follows by launching UDP probe packets.
UDS3lines	System	Displays unchannelized DS3 line information.
Uptime	Diagnostic	Displays how long the MAX TNT has been up since its last reset.
Userstat	System	Displays user-session status.
Version	System	Displays software version information.
View	System	Changes content of a status window.
Whoami	User	Displays current User profile name.
Write	Update	Writes a profile.

Displaying system and slot card uptime

The Uptime command reports how long the system and its individual cards have been up. The `slotLastChange` MIB object in the Ascend Enterprise MIB also enables network management stations to obtain uptime information.

The Uptime command uses the following syntax:

```
admin>help uptime
uptime usage: uptime [ [ -a ] | [ [ shelf ] slot ] ]
uptime                  display the TNT system uptime.
uptime slot             display the TNT slot card uptime.
uptime shelf slot       display the TNT slot card uptime.
uptime -a               display the uptime for all TNT slot cards.
uptime -?               display this usage message.
```

Without an argument, the command displays system uptime. But in the following example, the command displays the uptime for all slot cards in the UP state (cards that are not in the UP state are not reported):

```
admin> uptime -a
19:15:26
{ shelf-1 slot-1 }          8t1-card    9 days 01:05:40        7.0.0
{ shelf-1 slot-2 }       4ether-card    9 days 01:05:28        7.0.0
{ shelf-1 slot-3 }      192hdlc-card    9 days 01:04:02        7.0.0
{ shelf-1 slot-4 }  48modem-56k-card    9 days 01:03:40        7.0.0
{ shelf-1 slot-6 }      48modem-card    9 days 01:04:30        7.0.0
{ shelf-1 controller } shelf-controller 9 days 01:06:10    7.0.0
```

Uptime displays the current time (`19:15:26` in the preceding example), identifies the slot card, the software version running on the card, and displays the length of time the system has been up, in days followed by hours:minutes:seconds. The following example shows that a modem card in slot 2 has been up for 12 days, 1 hour, 5 minutes and 53 seconds:

```
admin>uptime 1 2

16:14:36

{ shelf-1 slot-2 }          8t1-card    9 days 00:33:53      7.0.0
```

Displaying the system version

Use the Version command to determine which system software version is installed. For example:

```
admin> version
Software version 7.0.0
```

You can also read the Software-Release parameter in the Slot-Info profile to display the engineering or candidate release number of the code image, if applicable. For example:

```
admin> get slot-info { 1 2 0}
[in SLOT-INFO/{ shelf-1 slot-2 0 }]
slot-address* = { shelf-1 slot-2 0 }
serial-number = 8168153
```

```
software-version =  7.0
software-revision =  0
software-level = ""
hardware-level = 0
software-release = ""
```

Displaying boot-loader version

The Boot-SR-Version parameter in the System profile displays the version of the current `tntsrb.bin` file (the boot-loader). The boot-loader updates the value of this parameter with its version at every system reset.

Viewing the factory configuration

The read-only Base profile displays the software versions, enabled features, network interfaces, and other system information. To view the Base profile, use the Get command. For example:

```
admin>get base
[in BASE]
shelf-number = 1
software-version = 7
software-revision = 0
software-level = b
manufacturer = dba-ascend-mfg
d-channel-enabled = yes
aim-enabled = yes
switched-enabled = yes
multi-rate-enabled = yes
t1-pri-conversion-enabled = yes
frame-relay-enabled = yes
maxlink-client-enabled = enabled
data-call-enabled = yes
r2-signaling-enabled = no
serial-number = 7050270
hardware-level = 0
countries-enabled = 511
domestic-enabled = yes
modem-dialout-enabled = yes
firewalls-enabled = no
network-management-enabled = no
phs-support = no
selectools-enabled = no
routing-protocols-disabled = no
tnt-adsl-restricted = no
tnt-sdsl-restricted = no
tnt-idsl-restricted = no
xcom-ss7 = disabled
ss7asg = disabled
atmp-enabled = enabled
l2tp-enabled = disabled
```

```
pptp-enabled = disabled
ipinip-enabled = disabled
```

The Base profile displays system information that is not modified across resets. These values are read from the system ROM, security PAL, and from the hardware assembly itself. (For information about the parameters, see the *MAX TNT Reference Guide*.)

Note: The shelf-number is always 1 in a single-shelf system. In a multi-shelf system, it must be unique for each shelf.

Setting the system name

The MAX TNT sends this name to callers whenever it establishes a PPP link. The name is not used in DNS lookups.

You specify the system name in the System profile. For example, to set the MAX TNT unit's system name to `tnt01`, proceed as follows:

```
admin> read system
SYSTEM read
admin> set name = tnt01
admin> write
SYSTEM written
```

Setting the system time and date

This section explains how to set the MAX TNT system clock. The MAX TNT can also use Simple Network Time Protocol (SNTP—described in RFC 1305) to set and maintain its system time by communicating with an SNTP server across an IP interface. For information about configuring the MAX TNT to use SNTP, see the *MAX TNT Network Configuration Guide*.

Use the Date command to set the system time and date if it is incorrect when the system initializes. To view the date and time, enter the Date command with no argument:

```
admin> date
Mon Nov 2 11:11:00 1998
```

To set it, append the current date and time to the Date command, in the following format:

```
yymmddhhmm
```

This format uses a two-digit number for each of the following settings: year, month, day, hour, and minute, in that order. For example:

```
admin> date 9811021743
Mon Nov  2 17:43:00 1998
```

In the year field, 00 - 89 represents years 2000 to 2089, and 90-99 represents years 1990 to 1999. For example, to set a date in the year 1999, proceed as in the following example:

```
admin> date 9910130029
Wed Oct 13  0:29:00 1999
```

To set a date in the year 2001, proceed as in the following example:
admin> **date 0110130029**

```
Sat Oct 13  0:29:00 2001
```

You can also Get the Timedate profile to view the information:

```
admin> get timedate
[in TIMEDATE]
time = { 17 43 34 }
date = { Monday November 2 1998 }
```

The Time and Date parameters in the Timedate profile cannot be set directly. To change their values, use the Date command as shown above.

Managing onboard NVRAM

The system configuration is stored in the onboard non volatile random access memory (NVRAM). Some error conditions might require that you clear the MAX TNT configuration and reboot. When you clear NVRAM, the system is reinitialized and comes up unconfigured, just as it was when you first installed it.

You can then restore the configuration from a recent backup (see “Backing up and restoring a configuration” on page 2-17).

Warning: Make sure you have a recent backup before using the NVRAM command.

To see how NVRAM is being used, enter the NVRAM command with the `-u` option:

```
admin> nvram -u
```

To clear NVRAM, restoring the unit to its initial, unconfigured state, enter the NVRAM command without specifying an option:

```
admin> nvram
```

To clear NVRAM and enter debug mode, use the `-t` option:

```
admin> nvram -t
```

Resetting the unit

When you reset the MAX TNT, the unit restarts and terminates all active connections. All users are logged out and the default security level, configured in the User-Profile parameter, is reactivated. In addition, a system reset can cause a WAN line to temporarily be shut down due to momentary loss of signaling or framing information.

To reset the unit, enter the Reset command:

```
admin> reset
```

During a reset, the MAX TNT runs its Power-On Self Test (POST), just as it would if the unit were power-cycled.

To reset the master shelf and all slaves in a multishelf system, append the `-a` option to the Reset command. For example, while logged into the master shelf, the following command resets all the shelves in a multishelf system:

```
admin> reset -a
```

Note that the `-a` flag is not valid on slave shelves.

Viewing clock-source information

If a line is specified as the clock-source, it can be used as the source of timing information for synchronous connections, so both the sending device and the receiving device can determine where one block of data ends and the next begins. If multiple T1 lines specify that they are the clock-source (the default configuration), you can assign clock-source priority among multiple T1 lines.

To view the clock-source statistics, enter the Clock-Source command:

```
admin> clock-source

Master: slot-1/1 line 3
Source List:
    Source: slot-1/1 Available      priority: 1
```

Sources with layer 2 up, which are preferred, are marked with an asterisk. For information about configuring the clock source, see the *MAX TNT Glossary*.

Using PCMCIA flash cards

Each MAX TNT shelf supports up to two PCMCIA flash-memory cards. The system comes with onboard NVRAM, and each flash card provides its own additional memory. At present, the flash cards contain code for the slot cards, the shelf-controller, and profiles. The system configuration is stored in the onboard NVRAM.

The PCMCIA slots on the shelf-controller are labeled 1 (the slot on top) and 2 (the slot below). See Figure 2-1.

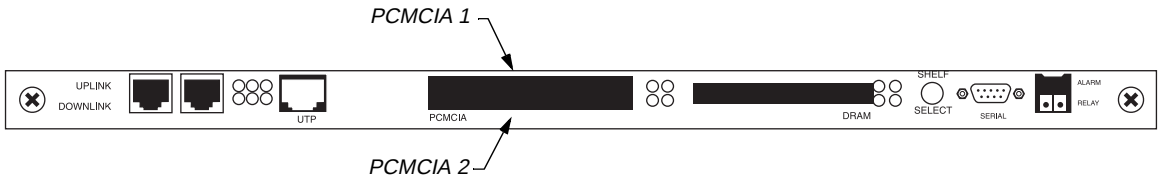

Figure 2-1. PCMCIA slots on the shelf-controller

Formatting a flash card

Before using a PCMCIA card in the MAX TNT, you must format it. First insert the card into slot 1 or slot 2 in the shelf-controller, then use the Format command. Following are examples of formatting the card in slot 1:

```
admin> format flash-card-1
```

Or:

```
admin> format 1
```

Flash-card-1 is the card inserted in the leftmost of the two PCMCIA slots.

For a list of error messages that might appear then using the Format command, see Appendix B, "MAX TNT Log Messages."

Displaying the contents of flash

The system comes with onboard NVRAM, and each flash card provides its own additional memory. The system configuration is stored in the onboard NVRAM.

To check the slot-card images stored in the flash card code directory, use the Dircode command, as shown in the following example:

```
admin> dircode
Flash card code directory:
Card 1, directory size 16
     shelf-controller reg   good  1237961 Nov  24 12:19      7.0
             8t1-card reg   good   203393 Nov  24 12:19      7.0
              t3-card reg   good   224951 Nov  24 12:19      7.0
          4ether-card reg   good   177007 Nov  24 12:19      7.0
           hdlc2-card reg   good   640052 Nov  24 12:19      7.0
           4swan-card reg   good   425375 Nov  24 12:19      7.0
    10-unchan-t1-card reg   good   510029 Nov  24 12:19      7.0
         ds3-atm-card reg   good   444831 Nov  24 12:19      7.0
            csmx-card reg   good   806361 Nov  24 12:20      7.0
```

The information displayed by this command includes the card number (1 or 2) and the size of the code directory. It also shows the following information about each code module:

- Type of card supported
- Subtype of the code, which can be regular or diagnostic
- Status, which can be good (present and complete), write (being copied), or bad (incomplete or corrupt)
- Size of the code
- Date the code was loaded to the flash card
- Code version

For a list of error messages that might appear when using the Dircode command, see Appendix B, "MAX TNT Log Messages."

Checking the file system

If the Dircode command shows a code status other than Good, or if you suspect inconsistencies in the flash card files, use the Fsck command to check the code directory. The Fsck command checks inconsistent conditions in the code directory as well as file contents on a PCMCIA flash card. For each file found, the command displays the type-name, type-number, decimal and hex byte counts, and date written to flash.

If errors are detected they are reported but not fixed. If the Fsck command reports errors, you should reformat the card and then load the code again. If necessary, download the code file again from the Ascend FTP server.

To check the file-system on the flash card in PCMCIA slot 1, use the Fsck command as shown in the following example:

```
admin>fsck 1
ffs check in progress for card 1...
Dir 1 not in use
Dir 2 has magic, version 2, size 16, sequence 0xa
Using dir entry: 2, total data blocks: 0x40, directory size: 16
shelf-controller:(0xfe)

    reg   good  1228008 (0x12bce8) Sep 23 18:08

8t1-card:(0x00)

    reg   good   195368 (0x02fb28) Sep 23 18:08

4ether-card:(0x10)

    reg   good   176597 (0x02b1d5) Sep 23 18:08

48modem-card:(0x01)

    reg   good   690472 (0x0a8928) Sep 23 18:09

t3-card:(0x06)

    reg   good   224620 (0x036d6c) Sep 23 18:09

4swan-card:(0x03)

    reg   good   423878 (0x0677c6) Sep 23 18:09

10-unchan-t1-card:(0x05)

    reg   good   508874 (0x07c3ca) Sep 23 18:09

hdlc2-card:(0x21)

    reg   good   637813 (0x09bb75) Sep 23 18:09

csmx-card:(0x31)

    reg   good   798139 (0x0c2dbb) Sep 23 18:10

flash card 1 fsck: good.
```

For details of the command-line options for the Fsck command, see the *MAX TNT Reference Guide*.

Updating system software

For information on updating system software, see the MAX TNT release notes.

Loading specific slot-card images

The MAX TNT supports a large number of slot cards, so the Tar files containing slot-card code images might be too large to load on an 8MB flash card. The Load-Select administrative profile enables you to specify which slot-card images to load to flash when you use a Load Tar command such as the one shown below:

```
admin> load tar network 10.10.10.10 tntrel.tar
```

Following a system reset, the MAX TNT creates the Load-Select profile if it is not present. The profile lists the entire set of supported slot-card images and an intended load action for each card type when the image is present in a Tar file. It also contains an Unknown-Cards parameter, which represents new cards that were not supported in the previous system version.

When loading the Tar file, the system uses settings in the Load-Select profile to load only specific slot-card images. To prevent version-related problems, it then deletes code images that were present on the flash card but were not updated.

For examples of upgrade procedures using the Load-Select profile, see the MAX TNT release notes.

Following are sample contents of the Load-Select profile:

```
[in LOAD-SELECT]
unknown-cards = auto
8t1 = auto
8e1 = auto
t3 = auto
ut1 = auto
ue1 = auto
uds3 = auto
ds3-atm = auto
enet = auto
enet2 = auto
mdm-v34 = auto
mdm56k = auto
amdm = auto
anmdm = auto
hdlc = auto
hdlc2 = auto
swan = auto
idsl = auto
capadsl = auto
dmtadsl = auto
sdsl = auto
sdsl70d = auto
sdsl70v = auto
```

Each parameter in the profile represents a card type, and can be set to Auto, Load, or Skip, to specify the action to take when the code image is present in a Tar file. (The Load-Select profile does not list the Shelf-Controller code, because that image is always loaded from the updated Tar file.)

- The Auto setting (the default) causes the system to load images for cards that are installed in the MAX TNT, and skip images for cards that are not installed. A card is considered present in the system if a Slot-Type profile exists for that card type. The system creates a Slot-Type profile when it first detects the presence of a card, and does not delete the profile unless the administrator uses the Slot –r command to permanently remove a card that is no longer installed in the system, or clears NVRAM. To ensure that the system does not load unnecessary images, use Slot –r to remove Slot-Type profiles for cards that are no longer installed in the system.
- The Load setting causes the system to load the image, even if there is no card of that type installed.
- The Skip setting causes the system to skip the image, even if there is a card of that type installed.

Loading images for unknown cards

The Unknown-Card type represents any card that was not known in the previous release. If a new card type is inserted into the system before loading the Tar file, the system loads all code images for all unknown cards, which could cause an overflow in the flash card. To prevent a flash overflow, use the following procedure:

1 Set the Unknown-Card field to Skip in the Load-Select profile.

```
admin> read load-select
LOAD-SELECT read

admin> set unknown-cards = skip

admin> write
LOAD-SELECT written
```

2 Save the current configuration to a TFTP server.

3 Load the new boot-loader from a TFTP server (or from the console). For example:

```
admin> load boot-sr network 10.10.10.10 tntsrb.bin
```

4 If you are upgrading a multishelf system, propagate the boot-loader to the slave shelves by using the Loadslave command.

5 Load the Tar file from a TFTP server (or from the console). For example:

```
admin> load tar network 10.10.10.10 tntrel.tar
```

6 Reset the system.

7 Insert the new card.

8 Load the Tar file again, to delete the images that are not required. For example:

```
admin> load tar network 10.10.10.10 tntrel.tar
```

Loading an extracted code image

You can override the settings in the Load-Select profile with options to the Load command. For example, if you extract the contents of a Tar archive and then issue the following Load command:

```
admin> load mdm56k network 10.10.10.10 tntmdm56k.ffs
```

The system loads the 56K-modem image even if the Load-Select profile indicates that it should be skipped. For details on the Load command, see the *MAX TNT Reference Guide*.

Backing up and restoring a configuration

The Save command saves all configured profiles, all profiles of a specified type, or a specific profile to a file on a local disk or to a file on a network host. You can then use that file to restore the MAX TNT configuration. Note that to save passwords, you must have sufficient permissions to view password fields (for a discussion of permissions, see "Understanding command permissions" on page 7-3).

Saving the configuration to a local file

To save the MAX TNT configuration to a file on the system you are using to access the MAX TNT, turn on the capture function in your VT100 emulation software, and enter the Save command as follows:

```
admin> save -a console
```

The entire configuration is written to the specified file. You might want to print a copy of the configuration for later reference.

The `-a` option saves all parameters, even those that are set to their default values.

Saving the configuration to a network host

To save the configuration on network host, you must specify the hostname and the full path of a filename, as in the following example:

```
admin> save -a network host1 /config/981001
configuration being saved to 10.65.212.19
```

In the sample command line, `host1` is the network host and `/config/981001` is the file name.

Restoring or updating the configuration

You can restore a full configuration that you saved with the Save command, or you can upload more specific configuration information, such as single profile.

To restore configuration information, use the Load command.

Restoring from a local file

Before you start the restore procedure, verify that your terminal emulation program has an autotype (or ASCII file upload) feature. Autotype allows your emulator to transmit a text file over its serial port. You should also verify that the data rate of your terminal emulation program is set to 9600 baud or lower and that the term-rate parameter in the System profile is

also set to 9600 or lower, and that the Term-Rate parameter in the System profile is set to the same rate. Speeds higher than 9600 baud might cause transmission errors.

To restore a configuration from a file on the system you are using to access the MAX TNT, set up your VT100 emulation software to send the file, and enter the Load command as follows:

```
admin> load config console
```

Restoring from a network host

To restore a configuration from a file on a network host, enter the Load command as follows:

```
admin> load config network hostname filename
```

Where `hostname` is the name of the host and `filename` is the name of the file in which the configuration is stored.

Updating the configuration

You can use the Load command to upload code for any of the slot cards to a flash card. For example, to upload new code for an eight port T1 card from a file named `8t1.ffs` on a network host named `server1`:

```
admin> load t1-8 network server1/cfg/8t1.ffs
```

Using the status window

The status windows provide information about what is currently happening in the MAX TNT. For example, one status window displays up to 31 of the most recent system events that have occurred since the MAX TNT was powered up, and another displays statistics about the currently active session. An 80-column by 24-row VT100 window is required for use of the status screens.

This section describes the default configuration of the Status windows. For information about customizing the status window display for User logins, see “Customizing the environment for a User profile” on page 7-6.

Status window command summary

By default, the status window is not displayed upon login, but only when you explicitly request it with one of the following commands:

- Status—Opens or closes the status window.
- Connection—Opens the status window with the connection information displayed.
- Line—Opens the status window with the line information display.
- Log—Opens the status window with the log information display.
- View—Changes the information displayed in the top or bottom status window.

For details of using these commands, see the *MAX TNT Reference Guide*.

Opening and closing the status window

To open the system status window, enter the Status command:

```
admin> status
```

The system prompt moves to just below the status window. If the system prompt is not visible below the status window, press Escape to display it.

To close the status window, enter the Status command again:

```
admin> status
```

Understanding the status window

The status window (Figure 2-2) has three main areas. In its default configuration, these areas contain the following information:

- Connections information is displayed on the left side of the window.
- General information, such as serial number, software version, and uptime are displayed in the upper-right side of the window.
- Log information is displayed in the lower-right side of the window.

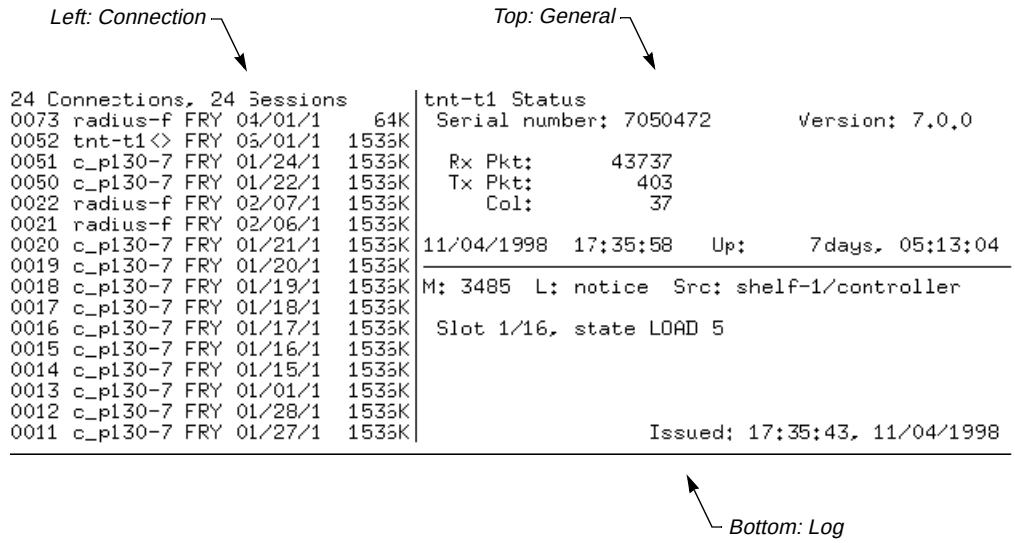

Figure 2-2. System status window

Connection status information

With the default setting in a User profile, the left area of the status window initially displays connection information, as shown in Figure 2-2. One line appears for each active connection, showing the user or station name, type of connection, T1 shelf, line, and channel on which the call was placed or received, and the bandwidth or baud rate of the connection.

If the status window is not already displayed, or if you want to scroll through the list of connections, use the Connection command as in the following example:

```
admin> connection
```

If the Status window is not displayed, the Connection command opens it and displays the connection-status-mode message below the Status window (if the Status window is already open, the Connection command just displays the message):

```
[Next/Last Conn:<dn/up arw>, Next/Last Page:<pg dn/up>,Exit: <esc>]
```

This message indicates the key sequences you can use for displaying additional information in the Connection status area. The Down Arrow and Up Arrow keys display the next and previous connection, respectively, in the list of active connections. The Page Down and Page Up keys display the list a screen at a time.

When the connection-status-mode message is displayed, the system prompt does not appear at the bottom of the window. Press the Escape key to exit this mode and return to the system prompt.

General status information

With the default setting in a User profile, the top area of the status window initially displays general status information about the MAX TNT, including its serial number, the version of system software it is running, and the number of packets transmitted and received. This area also shows the current system date and time and how long the system has been up.

If the top of the status window is displaying another kind of information, such as T1 line information, you can redisplay the general status information with the View command:

```
admin> view top general
```

Log messages

With the default setting in a User profile, the bottom area of the status window initially displays the most recent message from the MAX TNT log buffer. The number of system event messages stored in the log is set by the Save-Number parameter in the Log profile.

The first line of the event log window shows the log entry number (M: 00 through M: N, where N is set in the save-number parameter of the Log profile), the level of message, and the device on which the event occurred. The last line shows the date and time when the event occurred.

The middle of the window displays the text of the most recent message.

If the status window is not already displayed, or if you want to scroll through the log, use the Log command:

```
admin> log
```

If the Status window is not displayed, the Log command opens it and displays the log-mode message below the Status window (if the Status window is already open, the Log command just displays the message):

```
[Next/Last Conn:<dn/up arw>, Next/Last Page:<pg dn/up>,Exit: <esc>]
```

This message indicates the key sequences you can use for displaying additional information in the Log area:

- The Down Arrow and Up Arrow keys display the next and previous message in the buffer, respectively.
- The Page Up and Page Down. keys display the first and last message in the buffer, respectively.

When the log-mode message is displayed, the system prompt does not appear at the bottom of the window. Press the Escape key to exit this mode and return to the system prompt.

Displaying WAN line information

The status window can also display information about the WAN lines on the MAX TNT. For details, see “Displaying line status” on page 3-8.

Changing current status window sizes

The Screen command enables you to change the size of the terminal emulator and status windows for the current session. (For information about changing the terminal emulator and status windows for a User profile, see “Customizing the environment for a User profile” on page 7-6.)

The following command changes window display sizes for the current session only:

```
admin> screen screen-length [status-length]
```

If the Status window is open when you execute the Screen command, the Screen command resizes it dynamically. If it is not open, the Status window is resized when you next open it.

The *screen-length* option specifies the number of lines displayed in the terminal window. Note that *screen-length* must be at least 6 lines greater than the value of *status-length*.

The optional *status-length* option specifies the number of lines displayed in the status window, including dividing lines. The following example changes the terminal window to 55 lines high and the status windows to 22 lines high.

```
admin> screen 55 22
```

If you only specify the *screen-length* option, and it is not greater than the configured *status-length* by at least 6 lines, the MAX TNT automatically adjusts the length of the status windows. This is shown in the following example:

```
admin> screen 55 22
new screen-length 55
new status-length 22

admin> screen 24
error: screen-length conflict, adjusting status-length from 22 to 18
new screen-length 24
new status-length 18
```

Reviewing the fatal error log

The MAX TNT fatal error log contains messages related to the MAX TNT operations.

To view the log of fatal errors, enter the Fatal-History. For example:

```
admin> fatal-history
OPERATOR RESET:  Index:  99  Revision: 2.0     Shelf 1 (tntsr)
        Date: 01/30/1998.       Time: 16:55:38
        Reset from unknown, user profile admin.
  SYSTEM IS UP:  Index: 100  Revision: 2.0     Shelf 1 (tntsr)
        Date: 01/30/1998.       Time: 16:56:12
```

The command's output information includes the date and time at which the error occurred, the system software version that was running at that time, the slot number on which the error occurred, and a stack trace record of the event. (For a list of fatal error messages, see Appendix B, "MAX TNT Log Messages.")

To clear the fatal error log, enter the Clr-History command:

```
admin> clr-history
```

Configuring message logging

The MAX TNT generates error and event messages related to its operations. You can display these messages with the following commands:

- Log—Invoke or control the event log window.
- Fatal-History—List fatal error history log.

In the Log and User profiles you can configure the way in which the messages are handled .

The Log profile defines system-wide event logging parameters, including the number and level of messages to save and whether to communicate with a Syslog daemon.

Table 2-3 lists the sections describing common tasks you might have to perform to configure MAX TNT message logging. The table includes a brief description of each task, and lists the parameters you will use.

(For complete information about the associated parameters, see the *MAX TNT Reference Guide*.)

Table 2-3. Overview of configuring MAX TNT logging

Task	Description of task	Related parameters
"Configuring MAX TNT system logging"	You can configure the level and number of messages that are logged to the MAX TNT log. These messages are displayed in the log status window.	Save-Number Save-Level
"Configuring Syslog on the MAX TNT"	Syslog is a IP protocol that allows you to track events on the MAX TNT. A host running a Syslog daemon is typically a UNIX host, but it may also be a Windows system.	Sylsog-Enabled Call-Info Host Port Facility

Configuring MAX TNT system logging

The MAX TNT records system events in its status window event log. You can use the Save-Level and Save-Number parameters in the Log profile to configure the level and number of messages logged.

The Save-Level parameter specifies the lowest level of message to be saved for status display. The lowest possible level is None (this is the default). The highest level is Debug. For a list of the log message levels, see the *MAX TNT Reference Guide*.

The Save-Number parameter specifies the number of messages to be saved in the status display. The default is 100.

To configure the MAX TNT system log, proceed as in the following example:

1 Read in the Log profile:

```
admin> read log
LOG read
```

2 Specify the type of message you want logged:

```
admin> set save-level = emergency
```

3 Specify the number of messages to save in the event log:

```
admin> set save-number=200
```

4 Write the profile to save the changes:

```
admin> write
LOG written
```

Specifying a session ID base

The SessionID-Base parameter specifies the base number to use for generating a unique ID for each session. If SessionID-Base is zero, the MAX TNT sets the initial base for session IDs to the absolute clock. For details, see the *MAX TNT Reference Guide*.

Configuring Syslog on the MAX TNT

To maintain a permanent log of MAX TNT system events and send Call Detail Reporting (CDR) reports to a host that can record and process them, configure the MAX TNT to report events to a Syslog host on the local IP network.

The host running a Syslog daemon is typically a UNIX host, but it may also be a Windows system. If the log host is not on the same subnet as the MAX TNT, the MAX TNT must have a route to that host, either via RIP or a static route. (For information about Syslog messages, see "Syslog messages" on page B-7.)

Note: Do not configure the MAX TNT to send reports to a Syslog host that can only be reached by a dial-up connection. That would cause the MAX TNT to dial the log host for every logged action, including hang ups.

To configure Syslog, you might need to set some or all of the following parameters:

Parameter	Description
Sylsog-Enabled	Enables Syslog.
Call-Info	Specifies whether the MAX TNT sends a one-line Syslog message to the Syslog host when an authenticated call terminates. This message includes information such as the called and calling number and the encapsulation, data rate, and length of session.
Host	The IP address of the Syslog host.
Port	Specifies the port number on which the remote Syslog daemon is listening. It is set to port 514 by default.
Facility	Identifies the messages as being from a particular MAX TNT.
Syslog-Format	Specifies whether the messages the MAX TNT sends to Syslog are in MAX TNT format (the default) or in the same format as other Ascend MAX products.

To configure Syslog reporting on the MAX TNT, proceed as in the following example:

1 Read in the Log profile:

```
admin> read log
LOG read
```

2 Enable Syslog:

```
admin> set syslog-enabled = yes
```

3 Specify that you want end of call information sent:

```
admin> set call-info=end-of-call
```

4 Specify the IP address of the host running Syslog:

```
admin> set host=10.2.3.4
```

5 Specify the port the Syslog daemon is listening on:

```
admin> set port=588
```

The MAX TNT will send all messages out on this port as soon as you write the Log profile.

6 Specify the Syslog facility:

```
admin> set facility=local0
```

After setting a log facility number, you need to configure the Syslog daemon to write all messages containing that facility number to a particular log file. This file will be the MAX TNT log file.

7 Specify the format of Syslog messages:

```
admin> set syslog-format = max
```

8 Write the profile to save the changes:

```
admin> write
LOG written
```

Note that Call-Info is intended for diagnostic support. It uses UDP, which provides no guaranteed delivery, so it should not be used for billing purposes.

Configuring the Syslog daemon

To configure the Syslog daemon to interact with the MAX TNT, you need to modify the `/etc/syslog.conf` file on the log host. This file specifies which action the daemon will perform when it receives messages from a particular log facility number (which represents the MAX TNT). For example, if you set Log Facility to Local5 in the MAX TNT, and you want to log its messages in `/var/log/tnt01`, add the following line to `/etc/syslog.conf`:

```
local5.info<tab>/var/log/tnt01
```

Note: The Syslog daemon must reread `/etc/syslog.conf` after it has been changed.

Checking the power supplies

To check the status of the MAX TNT redundant power supplies, enter the Power command. For example:

```
admin> power
Power supply A not present
Power supply B present, OK
```

You can also use the Ascend Power Supply MIB to manage and monitor the power supplies.

Expanding system memory

If you need to accommodate very large routing tables or memory-intensive WAN interfaces, such as T3, you can increase the memory on a shelf controller by inserting a DRAM (JEDEC) card of 4, 8, 16, and 32MB.

To expand system memory:

1. Power down the MAX TNT.
2. Insert the DRAM card into the slot on the shelf controller labeled *DRAM*. (See Figure 2-3.)
3. Power up the MAX TNT.

Figure 2-3. DRAM slot

Using a script to configure the MAX TNT

The MAX TNT CLI allows you to create configuration scripts with a simple text editor and a Telnet client program with a Text Upload feature. This section briefly describes how you could use a script to make changes to the MAX TNT configuration.

Following are the basic steps:

1. Create a text file that contains the configuration commands as you would enter them in the MAX TNT CLI.
2. Log into the MAX TNT with sufficient permissions to change the configuration.
3. To upload the file to the MAX TNT, use the upload file feature of your Telnet of terminal software

Creating a text file

Following is an example of a text file that configures a T1 line in shelf 1, slot 1.

```
new T1
set name = SF
set physical-address shelf = shelf-1
set physical-address slot = slot-1
set physical-address item-number = 1
set line-interface enabled = yes
set line-interface frame-type = esf
set line-interface encoding = b8zs
set line-interface clock-source = eligible
```

```
write -f
;
```

Note: The Write -f command causes the script to overwrite an existing configuration without prompting.

You can use this file as a basis for configuring all twenty-eight lines on a DS3 card by changing the parameters, such as Item-Number, as required. Carefully review your text file to make sure it is correct.

Logging into the MAX TNT

To log into the MAX TNT for administrative tasks, use a profile that has write permissions, as in the following example:

```
% telnet mytnt
User: admin
Password: mypassword
admin>
```

If you are already logged into the MAX TNT, make sure you are at the highest level by entering the `list ..` command (possibly more than once), as in the following example:

```
admin>list ..
name = ""
physical-address* = { shelf-1 slot-1 1 }
line-interface = { yes esf b8zs eligible middle-priority
inband wink-start digi+

admin>list ..
error: at highest level
```

Uploading the text file

Use an ASCII text upload to upload the text file directly to the MAX TNT prompt. Carefully review your changes through the console.

Displaying user session information

You can obtain MAX TNT user session information with the Userstat and Finger commands.

Using the Userstat command

The Userstat command displays the active users on the MAX TNT. To display the most complete information about active sessions, use the `-l` option, as in the following example:

```
admin> userstat -l
SessionID  Line/Chan  Slot:Item  Tx/Rx Rate  Svc Address      Username
228687860  1.01.02/01 1:03:01/01 56K/56K     PPP 10.100.0.1   barney
228687861  1.02.03/02 1:04:02/00 28800/33600 PPP 10.168.6.24  jake
<end user list> 2 active user(s)
```

Following are the Userstat output fields with descriptions:

Field	Description
`SessionID`	Unique ID assigned to the session.
`Line/Chan`	Physical address (shelf.slot.line/channel) of the network port on which the connection was established, (for example, a T1 line/channel).
`Slot:Item`	*Shelf:slot:item/logical-item* of the host port to which the call was routed (for example, modem, HDLC channel).
`Tx/Rx Rate`	Transmit and receive rate. Note that for modem connections, the transmit rate is set automatically to the receive rate, because modem cards do not support asymmetric data rate connections.
`Svc`	Type of service in use for the session. Following are the possible values: --- (The service is being negotiated.) PPP (Point-to-Point Protocol) SLP (Serial Line IP) MPP (Multilink Protocol Plus) MP (Multilink Protocol) X25 (X.25) FRY (Frame Relay) EUR (EU-RAW) EUI (EU-UI) TLN (Telnet) BTN (Binary Telnet) TCP (raw TCP) TRM (Terminal Server) VCN (Virtual Connect) D25 (D-channel X.25) DTP (DTPT)
`Dialed#` (displays only with `-l` option)	The number dialed to initiate this session.
`ConnTime` (displays only with `-l` option)	The amount of time (in hours:minutes:seconds format) since the session was established.
`IdleTime` (displays only with `-l` option)	The amount of time (in hours:minutes:seconds format) since data was last transmitted across the connection.

To terminate a user, use the `-k` option, as in the following example:

```
admin> userstat
SessionID  Line/Chan  Slot:Item   Rate  Svc Address         Username
246986325 1.01.02/01 1:13:01/000 33600 PPP 100.100.8.2   100.100.8.2
<end user list>  1 active user(s)

admin> userstat -k 246986325
Session 246986325 cleared
```

The Userstat command can terminate PPP, SLIP, MP+, Telnet, Telnet binary, Raw TCP, or terminal server user sessions. You cannot use the `-k` option to terminate Frame Relay or DTPT

service types.

You can configure the Userstat command output with the Userstat-Format parameter. For information, see the *MAX TNT Reference Guide*.

Using the Finger command

Finger is described in RFC 1288. To enable it in the MAX TNT, set the Finger parameter to Yes, as follows:

1 Read the IP-Global profile:

```
admin> read ip-global
```

2 Set Finger to Yes:

```
admin> set finger = yes
```

3 Write the profile:

```
admin> write
```

The default value for this parameter is No, which causes the MAX TNT to reject queries from Finger clients with the following message:

```
Finger online user list denied.
```

Setting the Finger parameter to Yes enables the MAX TNT to accept Finger queries and return the requested active session details to a remote client. The client can ask for a short or wide format. For example, a UNIX client can request the wide (140-character) format by using the `-l` option, as in the following command which displays, in wide format, session information for the system named `tnt1`:

```
# finger -l @tnt1
```

The following command displays the same information in narrow (80-character) format:

```
# finger @tnt1
```

The client can also request the details of all sessions, or of a single session. For example, to request information about a single user named Gavin:

```
# finger gavin@tnt1
```

The Finger forwarding service, which uses the hostname format `@host1@host2`, is not supported. If the remote client uses the forwarding request format, the client sees the following message:

```
Finger forwarding service denied.
```

Call logging using the RADIUS accounting protocol

Call logging is a RADIUS-accounting based feature for logging call information from the MAX TNT. Its main purpose is to duplicate accounting information for sites that want to keep accounting records separate from other groups that might need call-logging details to manage resources or troubleshoot call problems.

Once you have configured call logging, the MAX TNT sends Start Session, Stop Session, and Failure-to-Start Session packets to a call-log host. A call-log host is a local host that supports the RADIUS accounting protocol and is configured properly to communicate with the MAX TNT (for example, a RADIUS accounting server or a host running NavisAccess). The call-log information is sent independently of RADIUS accounting records. If both call logging and RADIUS accounting are in use, the information is sent in parallel.

You set the following parameters, shown with their default values, to configure the MAX TNT to communicate with one or more call-log hosts:

```
CALL-LOGGING
   call-log-enable = no
   call-log-host-1 = 0.0.0.0
   call-log-host-2 = 0.0.0.0
   call-log-host-3 = 0.0.0.0
   call-log-port = 0
   call-log-key = ""
   call-log-timeout = 0
   call-log-id-base = acct-base-10
   call-log-reset-time = 0
   call-log-stop-only = yes
   call-log-limit-retry = 0
```

The parameters shown have the following functions:

Parameter	Function
Call-Log-Enable	Enables call logging. If set to No, none of the other call-logging parameters apply. If set to Yes, you must specify the IP address of at least one call-log host in the Call-Log-Host-*N* parameters
Call-Log-Host-*N*	Each specifies the IP address of one call-log host. The MAX TNT first tries to connect to server #1 for call-logging. If it receives no response, it tries to connect to server #2. If it receives no response from server #2, it tries server #3. If the MAX TNT connects to a server other than server #1, it continues to use that server until it fails to service requests, even if the first server has come online again.
Call-Log-Port	Specifies the UDP destination port to use for call-logging requests. The default value of 0 (zero) indicates any UDP port. If you specify a different number, the call-log host must specify the same port number (the numbers must match).
Call-Log-Key	A shared secret that enables the server to receive data from the MAX TNT. The value must match the configured shared secret on the call-log host.
Call-Log-Timeout	Specifies the number of seconds the MAX TNT waits for a response to a call-logging request. It can be set to a value of from 1 to 10. The default value is 0 (zero), which disables the timer.

Parameter	Function
Call-Log-ID-Base	Specifies whether the MAX TNT presents a session ID to the call-log host in base 10 or base 16. The default is base 10.
Call-Log-Reset-Time	Indicates the number of seconds that must elapse before the MAX TNT returns to using the primary call-log host (Call-Log-Host-1). The default value of 0 (zero) disables the reset to the primary call-log host.
Call-Log-Stop-Only	Specifies whether the MAX TNT should send an Stop packet with no user name. The MAX TNT typically sends Start and Stop packets to record connections. Authentication is required to send a Start packet. There are situations that the MAX TNT will send an Stop packet without having sent an Start packet in which case the Stop packets have no user name. The default value for Call-Log-Stop-Only is Yes. You can set it to No to prevent the unit from sending Stop packets with no user name.
Call-Log-Limit-Retry	If the server does not acknowledge a Start or Stop packet within the number of seconds specified in Call-Log-Timeout, the MAX TNT tries again, resending the packet until the server responds or the packet is dropped because the queue is full. The Call-Log-Limit-Retry parameter sets the maximum number of retries for these packets. The default value of 0 (zero) indicates an unlimited number of retries. There is minimum of 1 retry.

Following is an example of a procedure that enables call logging, specifies one call-log host on the local network, and specifies 10 retries:

```
admin> read call-logging
CALL-LOGGING read
admin> set call-log-enable = yes
admin> set call-log-host-1 = 10.2.3.4
admin> set call-log-limit-retry = 10
admin> write
CALL-LOGGING written
```

For complete information about the call logging parameters, see the *MAX TNT Reference Guide*. For information about configuring RADIUS, see the *MAX TNT RADIUS Configuration Guide*.

Reloading profiles from RADIUS

Use the Refresh command to open a connection to a RADIUS server and retrieve the latest configuration information. (For information about RADIUS, see the *MAX TNT RADIUS Configuration Guide*.)

The Refresh command uses the following syntax:

`refresh -a|-n|-p|-r|-t`

Option	Description
`-a`	Refresh all types of configuration.
`-n`	Refresh nailed profiles configuration.
`-p`	Refresh address pools configuration.
`-r`	Refresh static routes configuration.
`-t`	Refresh terminal server configuration.
`-s`	Clears the current Source Auth information (purging all existing Source Auth entries from the cache) and reloads it from RADIUS.

When you use the -n option, the MAX TNT requests a reload of all nailed profiles from the RADIUS server:

```
admin> refresh -n
```

You can specify how nailed connections are handled following a Refresh –n by using the Perm-Conn-Upd-Mode parameter in the System profile. If set to All (the default), all existing permanent connections are brought down and then brought up again (along with any new connections) following the update. This causes service interruption every time any nailed profile is updated or added.

If set to Changed, only new connections are created, and only those with modified attribute values are reestablished.

Configuring the dialout timer

The Max-Dialout-Time parameter in the System profile specifies the maximum number of seconds the system waits for a Call Setup Complete from the remote side when dialing out. If the MAX TNT cannot establish the call before the timer expires, the dialout attempt fails. The dialout timer allows increased flexibility for international dialing.

Valid values are from 0 to 255. The default is 20 seconds. If set to zero, the MAX TNT uses its internal default of 20 seconds. In the following example, the dialout timer is set to 60 seconds:

```
admin> read system
SYSTEM read

admin> set max-dialout-time = 60

admin> write
SYSTEM written
```

The Max-Dialout-Time setting does not influence the modem timeout to detect carrier. Modems have an internal timer that counts down from dialout to establishing carrier with the remote modem (including training) which for Rockwell modems has a default of 45 seconds.

Administering MAX TNT Slot Cards

3

This chapter covers the following topics:

Typical system administration tasks for the MAX TNT slot cards include viewing status information, removing a slot card configuration, and disabling lines. For information about managing the MAX TNT system, see Chapter 2, “MAX TNT System Administration.”

Viewing installed slot cards

The Show command displays information about the slot cards installed in the MAX TNT and the status of each card. In a multishelf system, the Show command displays cards in all shelves the system. You can also use the Show command for a particular slot card. For an example, see "Viewing information about a particular slot card" on page 3-3.

The following example illustrates use of the Show command to display a list of slot cards installed in a multishelf system:

```
admin> show
Shelf 1 ( master ):
    { shelf-1 slot-1 0 }       UP       4ether-card
    { shelf-1 slot-3 0 }       UP       48modem-card
    { shelf-1 slot-4 }         OCCUPIED
    { shelf-1 slot-15 0 }      UP       192hdlc-card
    { shelf-1 slot-16 0 }      UP       8t1-card
    { shelf-9 slot-1 0 }       UP       4ether-card
    { shelf-9 slot-2 0 }       UP       192hdlc-card
    { shelf-9 slot-15 0 }      UP       48modem-card
    { shelf-9 slot-16 }        OCCUPIED
    { shelf-9 controller 0 }   UP       shelf-controller
```

The output lists the physical address of each slot in which a slot card is installed. The address is in the form {*shelf slot item*}. Each listing also shows the status of the card and the type of card installed.

The status can be reported as follows:

Status	Signifies
UP	Normal operational mode.
DOWN	Not in an operational mode.
POST	The card is running power-on self tests.
LOAD	The card is loading code as part of booting up.
OCCUPIED	The slot is occupied by a two-slot card (such as the 48 modem card in shelf 1, slots 3 and 4, in the example above).
RESET	The card is being reset.
NONE	The card has been swapped out, but its configuration remains in NVRAM.

The Show command can report the following types of slot cards:

Label	Signifies
`unknown`	Current software does not recognize the card in the slot.
`10-unchan-t1-card`	T1 FrameLine card.
`10-unchan-e1-card`	E1 FrameLine card.
`24sdsl-data-card`	24 High Performance SDSL card.
`32idsl-card`	32-port IDSL card.
`4/1ether-card`	Ethernet card with one 100Mbps and four 10Mbps ports.
`4ether-card`	Ethernet card with four 10Mbps ports.
`4ether2-card`	Ethernet card with one 100Mbps and three 10Mbps ports.
`4swan-card`	4-interface serial WAN card.
`48modem-card`	48 V.34 modem card.
`48modem-56k-card`	Series56 Digital Modem card.
`8e1-card`	8-line E1 slot card.
`8t1-card`	8-line T1 slot card.
`analog-modem-card`	26 port analog modem card.
`capadsl-card`	6-port ADSL-CAP card.
`csmx-card`	Series56 II Digital Modem card.
`dmtadsl-card`	6-port ADSL-DMT card.
`ds3-atm-card`	DS3 card with ATM support.
`hdlc-card`	HDLC (Hybrid Access) card.
`hdlc2-card`	HDLC2 card.
`shelf-controller`	Shelf-controller card.
`sdsl-card`	16-port SDSL card.
`T3 card`	DS3 card.
`uds3-card`	Unchannelized DS3 card.

Viewing information about a particular slot card

To use the Show command for information about a particular command, add the shelf and slot-card numbers as arguments. For example:

```
admin>show 1 3
Shelf 1 ( standalone ):

    { shelf-1 slot-3 0 }       UP       4ether2-card:
    { shelf-1 slot-3 1 }       UP         ethernet-1
    { shelf-1 slot-3 2 }       UP         ethernet-2
    { shelf-1 slot-3 3 }       UP         ethernet-3
```

```
{ shelf-1 slot-3 4 }       UP         ethernet-4
{ shelf-1 slot-3 5 }                  100-Base-T
```

Opening a session with a slot card

To open a session with a slot card, use the Open command as in the following example:

```
admin> open 1 7
```

where 1 is the shelf number and 7 is the slot number.

On the master shelf of a multishelf system, you can open a session with a slave shelf. For example:

```
admin> open 3 17
```

You cannot use the Open command from the slave shelf.

After you have established a session with the card, the prompt changes to indicate the type of card, its slot number, and its shelf number. To list the commands available on the card, enter a `?` or `help`, as in the following example:

```
t1-1/1> ?
 ?                              ( user )
 auth                           ( user )
 cbcardif                       ( debug )
 cbsnmptrap                     ( debug )
 cbStats                        ( debug )
 checkd                         ( debug )
 clear                          ( user )
 clock-source                   ( diagnostic )
 debug                          ( diagnostic )
 debugd                         ( debug )
 display                        ( debug )
 dp-decode                      ( debug )
 dp-ram-display                 ( debug )
 dpram-test                     ( debug )
 dspBypassClients               ( debug )
 dspDial                        ( debug )
 dspSetDddTimeslot              ( debug )
 fakeCalledId                   ( debug )
 fakeClid                       ( debug )
 fe-loop                        ( diagnostic )
 fill                           ( debug )
 frreset                        ( debug )
[More? <ret>=next entry, <sp>=next page, <^C>=abort]
```

For information about the card-level commands, see the *MAX TNT Reference Guide*.

To exit the session with the card, enter `quit`, as in the following example:

```
t1-1/1> quit
```

Changing a slot state

To force a change in the state of a slot, use the Slot command, as shown in the following examples.

To bring a slot down, use the Slot command with the `-d` option, and specify the shelf and slot number of the card you want to down. For example:

```
admin> slot -d 1 3
slot 1/3 state change forced
```

When you bring a card down with the Slot command, it only remains down until the next reboot.

To bring a slot up:

```
admin> slot -u 1 3
slot 1/3 state change forced
```

You cannot change the state of a slave shelf controller by using the Slot –u or Slot –d commands.

Changing a device state

To force a change in the state of a device, use the Device command, as shown in the following examples.

To bring a device down:

```
admin> device -d {{1 3 6} 24}
slot 1/3 state change forced
```

To bring a device back up:

```
admin> device -u {{1 3 6} 24}
slot 1/3 state change forced
```

Removing a slot card and its configuration

MAX TNT slot cards are hot swapable. When you remove a card, the system retains its configuration. This enables you to re-install the card or install another of the same type in the same slot, without reconfiguring the system or uploading a backup configuration. One side-effect of this feature is that the NVRAM used to store configuration information is not cleared when a card is removed, until you explicitly clear the configuration.

When a card has been removed, it shows up with a status of NONE in the Show command output. For example:

```
admin> show 1 13
Shelf 1 ( master ):
  { shelf-1 slot-13 0 }      NONE      slot-card-8t1:
    { shelf-1 slot-13 1 }                  t1-line-1
    { shelf-1 slot-13 2 }                  t1-line-2
    { shelf-1 slot-13 3 }                  t1-line-3
```

```
{ shelf-1 slot-13 4 }              t1-line-4
{ shelf-1 slot-13 5 }              t1-line-5
{ shelf-1 slot-13 6 }              t1-line-6
{ shelf-1 slot-13 7 }              t1-line-7
{ shelf-1 slot-13 8 }              t1-line-8
```

The NONE status indicates that the card was removed but its profiles have been saved. The MAX TNT remembers that a card was in that slot and saves its profiles until a card of a different type is installed in the same slot, or until the administrator enters the Slot -r command, as in the following example:

```
admin> slot -r 13
slot 1/13 removed
```

In either case, all the old profiles associated with the slot are deleted. If a different type of card is inserted, appropriate new profiles are created.

Viewing the clock source for a slot card

The Clock-Source command can be run on the shelf controller or on an individual card, as in the following example on a T1 card:

```
admin> open 1 1

t1-1> clock-source
Master line: 3
Source List:
    Source: line 3 Available*       priority: 1
```

Sources with layer 2 up, which are preferred, are marked with an asterisk. For information about configuring the clock source see the *MAX TNT Glossary*.

Recovering from a failed slot-card installation

If you installed a new slot card before upgrading the system software, and the slot card does not come up properly, there are two ways to recover:

- Use the NVRAM command.
- Remove the slot card.

Using the NVRAM command

Warning: Using the NVRAM command resets the entire system. This method cannot be done remotely because the NVRAM command clears the MAX TNT configuration, including its IP address. Before performing this procedure make sure you have access to the MAX TNT serial port.

To recover from a failed slot-card installation by this method:

1 Save the current system configuration. For example:

```
admin>save network bonzo 971001
```

This saves the configuration to a file named 971001 in the TFTP home directory on a host named bonzo.

2 Clear the system configuration and restart the MAX TNT by executing the NVRAM command:

```
admin>nvramclear
```

3 Restore the saved system configuration.

You can either restore it through the serial port, or you can reassign an IP address and default gateway through the serial port, then use the Load command to load the rest of the configuration as in the following example:

```
admin>load config network bonzo 971001
```

This restores the configuration from a file named 971001 in the TFTP home directory on a host named bonzo.

For a complete description of saving and restoring configurations, see the "Backing up and restoring a configuration" on page 2-17.

Removing the slot card

To recover from a failed slot-card installation by removing the slot card:

1 Save the current configuration of any profiles on the card. For example:

```
admin>save network bonzo 971001 t1
```

This saves the configuration of all the T1 profiles to a file named 971001 in the TFTP home directory on a host named bonzo.

2 Bring down the card, as in the following example:

```
admin> slot -d 1 1
```

This disables the slot card in shelf 1, slot 1.

3 Remove the card profile:

```
admin> slot -r 1 1
```

4 Bring the card back up:

```
admin> slot -u 1 1
```

5 Restore the configuration of any profiles on the card. For the T1 card in this example, you would enter the following command:

```
admin>load config network bonzo 971001
```

This restores the configuration from a file named 971001 in the TFTP home directory on a host named bonzo.

Displaying line status

To display the activity of the MAX TNT WAN lines, enter the Line command:

```
admin> line [all|enabled] [top|bottom]
```

where

- `all` displays all lines.
- `enabled` displays enabled lines.
- `top` displays the status window at the top of the screen.
- `bottom` displays the status window at the bottom of the screen.

Figure 3-1 shows an example of a line-status window for the T3 card.

```
1 Connections, 1 Sessions       |"my T3"  1/15/00 LA la  la  la  la  la  la  la
0065 FRM2-SLC MPP 09/02/1  56000|         1/15/01 LA T-------- -------- --------
                                |         1/15/02 LA T-------- -------- --------
                                |         1/15/03 LA T-------- -------- --------
                                |         1/15/04 LA T-------- -------- --------
                                |         1/15/05 LA T-------- -------- --------
                                |         1/15/06 LA T-------- -------- --------
                                |         1/15/07 LA T-------- -------- --------
                                |-----------------------------------------------
                                |M: 520  L: notice  Src: shelf-1/slot-15
                                |
                                | Line 28 up
                                |
                                |
                                |
                                |
--------------------------------------------------------------------------------
[ Next/Last Line: <up/dn arw>, Next/Last Page: <pg up/dn>, Exit: <esc> ]
```

Figure 3-1. Example of a T3 card line-status window

The first entry in the right-hand area of the screen shows the overall status of the DS3 line and each of its seven component DS2 channels. One DS2 includes 4 DS1s. The other entries represent each of the component DS1s.

The Line commands put the window in line-status mode, in which the following message appears below the status window:

```
[Next/Last Conn:<dn/up arw>, Next/Last Page:<pg dn/up>,Exit: <esc>]
```

The message indicates the key sequences you can use for displaying additional information in the line status area. The Down Arrow and Up Arrow keys display the next and previous T1 line in the list, respectively. The Page Down and Page Up keys display the list a screen at a time.

When the line-status mode message is displayed, the system prompt does not appear at the bottom of the window. Press the Escape key to exit this mode and return to the system prompt.

Line status information includes the following identifiers and codes:

- Line identifier in shelf/slot/line format
- Two-character code indicating the line's link status
- Single-character code indicating channel status
- Single-character code indicating channel type

Following are the link-status codes:

Code	Description
`LA` (link active)	The line is active and physically connected
`LS` (UDS3 lines)	Loss of Signal. No signal has been detected.
`LF` (UDS3 lines)	Loss of Frame. A signal is present but is not valid for framing.
`NT`	The E1 line is active and configured as network-side equipment.
`TE`	The E1 line is active and configured as user-side equipment.
`RA` (red alarm)	The line is unconnected, improperly configured, experiencing a very high error rate, experiencing a loss-of-receive-signal, or is not supplying adequate synchronization.
`YA` (yellow alarm)	The MAX TNT is receiving a Yellow Alarm pattern, an indication that the other end of the line cannot recognize the signals the MAX TNT is transmitting.
`DF` (d-channel fail)	The D channel for a PRI line is not currently communicating.
`1S` (all ones)	A keep-alive (also known as a Blue Alarm) signal is being sent from the PRI network to the MAX TNT to indicate that the line is currently inoperative.
`ID` (idle—DS3 only)	The DS3 interface has detected an Idle Signal transmitted from the other side. This generally indicates that the line is provisioned but is not in use.
`WF` (wrong framing—DS3 only)	The DS3 interface has detected that the other side is using a framing format that differs from the one the local DS3 interface is configured for (C-bit-parity or M13).

Following are the channel-status codes:

Code:	Description
`.` (period)	The channel is not available because of one of the following reasons: • Line is disabled • Channel has no physical link • Channel does not exist • Channel configuration specifies that it is unused • Channel is reserved for framing (first E1 channel only)
* (asterisk)	The channel is connected in a current call.
- (hyphen)	The channel is currently idle (but in service).
b	The channel is a backup NFAS D channel (T1 PRI only).

Code:	Description
c	The channel is currently not available because it is in the process of clearing the most recent call, or because it is in the process of sending echo cancellation tones to receive a call (inband signaling on T1 only).
d	The MAX TNT is dialing from this channel for an outgoing call.
r	The channel is ringing for an incoming call.
m	The channel is in maintenance/backup mode (ISDN and SS7 only).
n	The channel is nailed.
o	The channel is out of service (ISDN and SS7 only).
s	The channel is an active D channel (ISDN only).

Following are the channel-type codes:

Code	Description
E	E1 line
I	T1 PRI signaling
N	All other NFAS types
P	NFAS Primary
S	NFAS Secondary
T	T1 inband signaling

Administering ATM DS3 cards

The DS3ATMlines, Framer, and ATMDumpCall commands allow you to perform diagnostics on the ATM DS3 card.

Using the DS3ATMlines command

This command uses the following syntax:

```
admin> ds3atmlines -option
```

where **-option** may be one of the following:

Option	Effect
-a	Displays all available ATM DS3 lines.
-d	Displays disabled ATM DS3 lines.
-f	Displays free ATM DS3 lines.
-u	Displays in-use ATM DS3 lines.

In the following example, the ATM DS3lines command displays all ATM DS3 lines:

```
admin> ds3atmlines -a
All DS3_ATM lines:
                                (dvOp   dvUpSt  dvRq    sAdm    nailg)
  Line    {    1  4  1 }       (Up      Idle    UP      UP     00000)
```

Regardless of which option you enter, the ATM DS3lines command displays the following information:

Column Name	Description
dvOp	The operational state of the DS3 line. Values can be: • Down • Up
dvUpSt	The up status of the DS3 line. Values can be: • Idle • Reserved • Assigned
dvRq	The required state of the DS3 line. Values can be: • Down • Up
SAdm	The desired state of the device. Values can be: • Down • Up
nailg	The nailed group that this line is assigned to.

Using the Framer command

The Framer command is a low-level management tool for use during diagnostic sessions with the ATM DS3 card. For example, to use the Framer command on a DS3 card on shelf 1 in slot 3, first enter the Open command as follows:

```
admin> open 1 3
```

Then, enter the Framer command:

```
ds3-atm-1/3> framer -option
```

where ***-option*** is one of the following:

Option	Effect
-t	Toggles debug output.
-d	Dump ATM framer chip status information. The information this command displays is also available from the status lights on the card and in the DS3-ATM-Stat profile.
-l	Toggle a local loopback.

Option	Effect
`-r`	Toggle a remote loopback.
`-s`	Synchronize to the ATM DS3 profile. The MAX TNT automatically re-reads the line configuration whenever it comes up.
`-c`	Clear the error counters.
`-?`	Displays this summary.

For example, to view overall status information about the ATM DS3 line, enter the Framer command with the `-d` option:

```
ds3-atm-1/4> framer -d
Framer is Enabled

RED_ALARM_LED   : Off
YELLOW_ALARM_LED: Off
AIS_LED         : Off
OOF_LED         : Off
ACTIVE_LED      : On


F-Bit  Error Counter: 35
P-Bit  Error Counter: 20
C-PBit Error Counter: 10
FEB    Error Counter: 51
BPV    Error Counter: 12
EZD    Error Counter: 39
```

Following are the Framer command output fields with descriptions:

State	Description
Red_Alarm_LED	On indicates the line is not connected, or it is improperly configured, experiencing a very high error rate, or supplying inadequate synchronization.
Yellow_Alarm_LED	On indicates the card is receiving yellow-alarm from far end.
AIS_LED	On indicates the card is receiving alarm indication signal.
OOF_LED	On indicates the near end is in an out of frame condition.
Active_LED	On indicates multipoint established.

The remaining parameters indicate the errors on the DS3 line. (Refer to RFC 1407 for complete description of these errors.)

Parameter	Description
F Bit Error Counter	Framing bit errors received since the last MAX TNT reset or the error counters were cleared.
P Bit Error Counter	P-bit errors indicate that MAX TNT received a P-bit code on the DS3 M-frame that differs from the locally calculated code.

Parameter	Description
CP Bit Error Counter	For C-Bit-Parity lines indicates that number of parity errors since the last MAX TNT reset.
FEB Error Counter	Far end block errors received since the last MAX TNT reset.
BPV Error Count	Bipolar Violation (BPV) errors may indicate that the line sent consecutive one bits with the same polarity. It could also mean that three or more consecutive zeroes were sent or an incorrect polarity.
EZD Error Counter	Number of Excessive Zero Detect (EZD) line code violations that have occurred since the error counters were cleared.

Using the ATMDumpCall command

The ATMDumpCall command is a low-level management tool for use during diagnostic sessions with the ATM DS3 card. It allows you to view the ATM call blocks, which contain information about outgoing calls.

For example, to manage a DS3 card on the shelf 1 in slot 3, first enter the Open command as follows:

```
admin> open 1 3
```

Then, enter the ATMDumpCall command:

```
ds3-atm-1/3> atmdumpcall -option
```

where ***-option*** is one of the following:

Option	Effect
`-a`	Display all ATM call blocks, even those that are inactive.
`-l`	Display DS3 ATM line configuration information.
`-u`	Display in-use ATM call blocks.

For example, to view all ATM call blocks, enter the ATMDumpCall command with the `-a` option:

```
ds3-atm-1/3> atmdumpcall -a
atmdumpcall -a
ATM Call Block Table:
Addr.     Index Active  callID  routeID State     Vpi/Vci     Prof_Name  Sess_Up
E00C47F0  0     1       1       1       CONNECTED 1/43        atm-30-sw  Yes
E00C4834  1     1       2       2       CONNECTED 15/1023     Yossi-TNT  Yes
E00C4878  2     1       3       3       CONNECTED 1/56        Yoss-P220  Yes
E00C48BC  3     0       65535   0       INACTIVE  0/0           -        No
E00C4900  4     0       65535   0       INACTIVE  0/0           -        No
.
.
.
E00C5868  62    0       65535   0       INACTIVE  0/0           -        No
```

```
E00C58AC  63    0       65535   0       INACTIVE  0/0           -        No

ATM Free Blocks: 360
ATM Used Blocks: 0
```

Using the OAMLoop command

The OAMLoop command sends ATM Operation-And-Maintenance (OAM) loop-back cells on an ATM interface, to obtain information about the results of the looped cells. It uses the following syntax:

admin> **oamloop -*option***

where option is one of the following:

Option	Description
`-e`	(End-to-End). Transmit an end-to-end OAM loop cell, to be looped by the user connection point. This option and the `-s` option are mutually exclusive, and one of them must be specified on the command line.
`-s`	(Segment). Transmit a segment OAM loop cell, to be looped by the first network connection point. This option and the `-e` option are mutually exclusive, and one of them must be specified on the command line.
`–c` *count*	Transmit the specified number of cells. If this argument is not specified, the count defaults to 0, which means that the cells are transmitted continuously until the administrator sends an interrupt by pressing Ctrl-C.
`–i sec`	Transmit the cells at the specified interval in seconds. If this argument is not specified, the interval defaults to one second.
shelf	Specifies the shelf in which the DS3 ATM card is located.
slot	Specifies the slot in which the DS3 ATM card is located.
vpi	Specifies the Virtual Path Identifier on which to transmit the looped-back cells.
vci	Specifies the Virtual Channel Identifier on which to send the looped-back cells.

Following is an example OAMloop command line and output:

```
admin> oamloop -c 10 -e 1 2 1 32
Received our End2End OAM loopback cell, Id=9
Received our End2End OAM loopback cell, Id=10
Received our End2End OAM loopback cell, Id=11
Received our End2End OAM loopback cell, Id=12
Received our End2End OAM loopback cell, Id=13
Received our End2End OAM loopback cell, Id=14
Received our End2End OAM loopback cell, Id=15
Received our End2End OAM loopback cell, Id=16
Received our End2End OAM loopback cell, Id=17
```

```
Received our End2End OAM loopback cell, Id=18
--- OAM loop statistics ---
10 cells transmitted, 10 cells received, 0% cell loss
```

Looping back the ATM DS3 line

For diagnostics, you might want to loopback the DS3 interface by using the Loopback parameter in the DS3-ATM profile. While the interface is looped back, normal data traffic is interrupted. The Loopback parameter in the DS3-ATM profile supports the following settings:

- No-Loopback. The default, specifies that the DS3 line is operating normally.
- Facility-Loopback. During a facility loopback, the DS3 card returns the signal it receives on the DS3 line.
- Local-Loopback. During a local loopback, the DS3 receive path is connected to the DS3 transmit path at the D3 multiplexer. The transmitted DS3 signal is still sent to the network as well.

Line statistics are displayed in the DS3-ATM-Stat profile. For information about this profile, see “Using DS3-ATM-Stat profiles” on page 9-15.

To configure a loopback, proceed as in the following example:

1 Read the DS3-ATM profile:

```
admin> read ds3-atm {1 3 1}
DS3-ATM/{ shelf-1 slot-3 1 } read
```

2 Activate the loopback:

```
admin> set line loopback= [facility-loopback|local-loopback]
```

3 To end the loopback, set the Loopback parameter to No-Loopback:

```
admin> set line loopback = no-loopback
```

Administering Ethernet cards

For all Ethernet interfaces except the shelf controller, the MAX TNT detects and flags changes in the interface link-state. You can enable a feature in the Ethernet profile that causes automatic routing table updates based on physical link-state changes. Routes to a disabled (down) interface are deleted from the IP routing table, so alternative configured routes can be used instead, and the routes are added again when the interface comes back up. You can also choose to administratively down a LAN interface by disabling its Ethernet profile.

The following parameters, shown with their default settings, are related to LAN-interface link-state changes:

```
ETHERNET {shelf-N slot-N item-N}
   enabled = yes
   link-state = up
   link-state-enabled = no
```

For information about configuring a management-only Ethernet interface, see the *MAX TNT Hardware Installation Guide*.

Enabling or disabling an Ethernet interface

The Enabled parameter in an Ethernet profile specifies whether a LAN interface is enabled (the default) or disabled. If Enabled is set to No, packets routed to and received on the interface are discarded. Note that the user-specified state is preserved across system resets.

An interface may also be disabled by using the Ifmgr command, or it may be marked as down by the Ethernet driver when Link-State-Enabled is Yes and Link-State is Down.

To enable an interface, set the Enabled parameter to Yes (the default), or use the Ifmgr Up option. Note, however, that if there are physical problems with the interface, specifying the interface as up might not enable it.

To disable an interface with the Ifmgr command, proceed as in the following example:

1 Open a session with an Ethernet card:

```
admin> open 1 4
ether-1/4> ifmgr
```

2 View the interface table:

```
ether-1/4> ifmgr -d
if  slot:if  u p ifname   mac addr      local-addr
-------------------------------------------------------------
000 0:00:000 *   pb0      000000000000 0.0.0.0/32
001 1:17:011 *   ie1-4-1  00c07b6d23f0 11.1.1.1/32
002 1:17:013 *   ie1-4-2  00c07b6d23f1 11.1.2.1/32
003 1:17:015 *   ie1-4-3  00c07b6d23f2 11.1.3.1/32
004 1:17:017 *   ie1-4-4  00c07b6d23f3 11.1.4.1/32
005 1:17:019 *   ie1-4-5  00c07b6d23f4 11.1.5.1/32
<end>
```

3 Mark the interface as down by specifying its name:

```
ether-1/4> ifmgr down ie1-4-1
```

The Ifmgr display indicates that the interface is disabled by displaying a dash instead of an asterisk in the Up column (u):

```
ether-1/4> ifmgr -d
if  slot:if  u p ifname   mac addr      local-add
-------------------------------------------------------------------
000 0:00:000 *   pb0      000000000000 0.0.0.0/32
001 1:17:011 -   ie1-4-1  00c07b6d23f0 0.0.0.0/32
002 1:17:013 *   ie1-4-2  00c07b6d23f1 11.1.2.1/32
003 1:17:015 *   ie1-4-3  00c07b6d23f2 11.1.3.1/32
004 1:17:017 *   ie1-4-4  00c07b6d23f3 11.1.4.1/32
005 1:17:019 *   ie1-4-5  00c07b6d23f4 11.1.5.1/32
<end>
```

Note: A disabled Ethernet interface is also shown with a dash in Netstat command output.

To mark an interface as up, enter a command similar to the following:

```
ether-1/4> ifmgr up ie1-4-1
```

For more information about the Ifmgr command, see “IFMgr” on page 5-18.

Specifying how the link state affects the IP routing table

The Link-State-Enabled parameter signifies whether the value of the Link-State parameter affects the IP routing tables. If it is set to Yes, routes to an interface are deleted when the link state is down, and added back when the interface comes back up again. If the parameter is set to No (the default), packets are routed to the interface regardless of its link-state. If the interface is down, packets are discarded rather than transmitted over using an alternative route.

A read-only indication of physical link-state

The Link-State parameter shows the physical state of the LAN interface: up or down. The parameter can only be set by the Ethernet driver. A LAN interface is down if it cannot transmit or receive network traffic (for example, if the Ethernet cable is unplugged or the Ethernet hub on that interface is down). For the shelf-controller Ethernet interface, the value of the Link-State parameter is set to Unknown.

Checking multiple IP interfaces on an Ethernet port

In the following Ifmgr command output, the physical interface 1-12-1 has two IP-Interface profiles associated with it. The first is named `ie1-12-1` (the default profile), and the second is named `ie1-12-1-1`:

```
admin> ifmgr -d
bif slot sif u m p ifname     host-name  remote-addr       local-addr
-------------------------------------------------------------------------------
000 1:17 000 *     ie0        -          0.0.0.0/32        200.168.6.188/32
001 1:17 001 *     lo0        -          0.0.0.0/32        128.0.0.1/32
002 0:00 000 *     rj0        -          0.0.0.0/32        128.0.0.2/32
003 0:00 000 *     bh0        -          0.0.0.0/32        128.0.0.3/32
004 0:00 000 *     local      -          0.0.0.0/32        128.0.0.1/32
005 0:00 000 *     mcast      -          0.0.0.0/32        225.0.0.0/32
006 1:12 001 *     ie1-12-1   -          0.0.0.0/32        10.5.6.7/32
007 1:12 002 *     ie1-12-2   -          0.0.0.0/32        0.0.0.0/32
008 1:12 003 *     ie1-12-3   -          0.0.0.0/32        0.0.0.0/32
009 1:12 004 *     ie1-12-4   -          0.0.0.0/32        0.0.0.0/32
010 1:12 005 *     ie1-12-1-1 -          0.0.0.0/32        10.9.1.212./24
```

Administering T1, T3, and T1 FrameLine cards

MAX TNT T1, T3, and FrameLine cards are all administered in much the same way. In most cases, administration of the individual T1 lines on the three cards is identical. Table 3-1 briefly describes the different methods you can use to manage the T1, T3, or FrameLine cards, and shows where each method is discussed in this manual.

Table 3-1. T1-line maintenance tasks

Task/section of this manual	Description	Associated parameter or command
"Quiescing a PRI line or T1 channels" on page 3-18.	Quiescing a PRI line allows you to gradually take a line or channels out of service.	Maintenance-State parameter Quiesce command
"Specifying FDL" on page 3-19.	Your T1 service provider can use Facilities Data Link (FDL) to monitor the status of your line.	FDL parameter
"Checking the status of T1 channels" on page 3-20.	Display the administrative state and nailed-group assignment of the T1 channels.	T1channel command
"Displaying DS1-level diagnostics for T1 cards" on page 3-20.	Display T1 channel errors.	T1-Stats command
"The FE-Loop command" on page 3-22.	Loopback the T1 line.	FE-Loop

Quiescing a PRI line or T1 channels

Quiescing a PRI line takes the line out of service by removing channels from service as active calls disconnect. The switch used by the carrier affects whether the line is taken out of service or busied out. For details, see the Quiesce command description in the *MAX TNT Reference Guide*.

You can quiesce a line by using either of the following methods:

- Maintenance-State parameter in the T1 profile
- Quiesce command

Restoring a line or channel that has been quiesced can take up to 10 minutes.

Note: You cannot quiesce a T1 line on the FrameLine card.

Using the Maintenance-State parameter

To quiesce a line with the Maintenance-State parameter, proceed as in the following example:

```
admin> set line maintenance-state=yes
admin> write
T1/{ shelf-1 slot-2 1 } written
```

Using the Quiesce command

You can enter the Quiesce command to quiesce a PRI line, port, or channel. The command uses the following syntax:

```
admin>quiesce -d|e|r|q|t line
```

where

- `-d` quiesces a single DS0 channel.
- `-e` restores a quiesced DS0 channel.
- `-r` *line* restores the quiesced line.
- `-q` *line* quiesces a PRI line.
- `-t` toggles the diagnostic display.

For example, to quiesce a T1 PRI line at port 4 of a card installed in slot 2:

```
admin> quiesce -q {1 2 4}
QUIESCE: line 1/2/4, enable=T, isPri=T
```

Restoring a line or channel that has been quiesced can take up to 3.5 minutes, because only 1 service message per channel is sent to the switch, at a rate of one per second.

To restore the line quiesced in the preceding example:

```
admin> quiesce -r {1 2 4}
QUIESCE: line 1/2/4, enable=T, isPri=T
```

Following is an example of quiescing a single channel:

```
admin> quiesce -d {{1 2 4} 1}
```

Specifying FDL

The facilities data link (FDL) is used by the telephone company to monitor the quality and performance of T1 lines. If your carrier's maintenance devices require regular data-link reports, and if the line is not configured for D4 framing, you can specify the type of protocol to use (AT&T, ANSI, or Sprint).

You cannot use FDL reporting on a line configured for D4 framing. However, you can obtain D4 and ESF performance statistics in the FDL Stats windows or the DSX MIB, even if you do not choose an FDL protocol. (For further information, see the Frame-Type parameter description in the *MAX TNT Reference Guide*).

Note: FDL is not supported on the FrameLine card. Also, DS3-level FDL capabilities such as the Far-End Alarm and Control Channel (FEAC) and Path Maintenance Data Link are currently unsupported.

To specify the type of FDL, proceed as in the following example:

```
admin> read t1 {1 2 1}
T1/{ shelf-1 slot-2 1 } read
admin> set fdl = [none|at&t|ansi|sprint]
admin> write
```

Checking the status of T1 channels

To show T1-channel information, enter the T1Channels command. Use the following syntax:

```
admin> t1channels - a|d|c|i
```

where

- **-a** displays all available channels.
- **-d** displays the disabled channels.
- **-c** displays all possible channels.
- **-i** displays in-use channels.

For example, to display all T1 channels available, use the -a option:

```
admin> t1channels -a

T1 channels available for use:
                                    (dvOp  dvUpSt  dvRq  sAdm nailg)
  Channel { {  1  1  3 }  1 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  2 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  3 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  4 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  5 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  6 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  7 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  8 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 }  9 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 } 10 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 } 11 }   (Up    Idle    UP    UP   00000)
  Channel { {  1  1  3 } 12 }   (Up    Idle    UP    UP   00000)
```

To display information about which T1 channels are in use:

```
admin> t1channels -i

T1 channels allocated/in-use:
                                    (dvOp  dvUpSt  dvRq  sAdm nailg)
  Channel { {  1  1  1 }  1 }   (Up    Assign  UP    UP   00000) I
  Channel { {  1  1  1 }  9 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 10 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 11 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 12 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 13 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 14 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 15 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1  1  1 } 16 }   (Up    Assign  UP    UP   00006) I
  Channel { {  1 10 10 }  1 }   (Up    Assign  UP    UP   00005) I
```

Displaying DS1-level diagnostics for T1 cards

The T1-Stats command reports DS1-level line errors. Before entering the command, use the Open command to open a session with the installed card. For example, to open a session with a card in shelf 1, slot 13:

```
admin> open 1 13
```

Then enter the T1-Stats command. The following example shows the command's syntax:

```
t1-1/13> t1-stats
t1-stats [ -c ] <line>  get error statistics for the line
    -c: reset statistics to zero
```

To view DS1-level statistics on the first line on the card:

```
t1-1/13> t1-stats 1
  Line 1:
CRC Errors:           0
Frame Slips:          8
Framing Bit Errors:   0
Out of Frame Events:  0
Line Code Violations: 0
```

Table 3-2 explains the T1-Stats fields.

Table 3-2. T1-Stats command fields

Field	Event that increments the field
CRC Errors	Indicates that a CRC-6 checksum shows data corruption in the signal.
Frame Slips	The MAX TNT receives T1 data at a frequency higher or lower than the internal line clock. In the process of realigning itself to the transmitter, the MAX TNT can skip or repeat a frame.
Framing Bit Errors	Framing bit errors occur when the MAX TNT receives T1 data at a frequency higher or lower than that of the internal line clock. In the process of realigning itself to the transmitter, the MAX TNT can skip or repeat a frame.
Out of Frame Events	The MAX TNT no longer detects a framing pattern in the receiving signal, or it detects a pattern at a different relative offset than expected.
Line Code Violations	The MAX TNT detected either a Bipolar Violation or Excessive Zeros, which means that one of the low-level T1 rules for encoding data was violated in the received signal.

The following example shows how to view and reset the statistics to zero on line 2:

```
t1-1/13> t1-stats -c 2
  Line 2:
CRC Errors:           2
Frame Slips:          3
Framing Bit Errors:   0
Out of Frame Events:  0
Line Code Violations: 3
  Statistics cleared.
```

The `Statistics cleared` message at the end of the display indicates that the statistics have been reset to 0 (zero), because the command included the `-c` option.

The FE-Loop command

When a T1 line is looped back to the network, either as a result of the FE-Loop diagnostic command issued from the T1 card command line interface or as a result of loopback requests received from the network, the T1 line status display on the shelf controller shows the `LB` (loopback) status for the line.

The following examples demonstrate the use of the FE-Loop command:

To verify that the hardware is functioning properly, perform a local loopback by using the FE-Loop command with the `in` option. For example, to internally loop back the first DS1 in slot 1:

```
admin> open 1 1
t1-1/1>
t1-1/1> fe-loop 1 in on
```

You can use this command when the line is in RA state. After looping back the line internally, state should change to LA. Note that the T3 card does not support the FE-Loop `in` option.

To turn the internal loopback off:

```
t1-1/1> fe-loop 1 in off
```

To cause the unit to transmit the received signal back towards the network, enter the following command:

```
t1-1/1> fe-loop 1 out on
```

The receive side of the T1 is not bridged to the MAX TNT. This command can be useful in testing the path to the MAX TNT by:

- Verifying that the switch can synchronize to its own returned signal.
- Supporting test equipment that sends out a test pattern, such as a Quasi-Random Signal (QRS), and verifying that the pattern is received unmodified.

To turn the remote loopback off:

```
t1-1/1> fe-loop 1 out off
```

Using DS3 diagnostics

The DS3Link command is a low-level management tool for use during diagnostic sessions with the T3 card. To open a session with the installed DS3 card, use the Open command.

For example, to manage a DS3 card on the shelf 1 in slot 15, first enter the Open command as follows:

```
admin> open 1 15
```

Then, enter the DS3Link command:

```
t3-1/15> ds3link -option
```

where ***-option*** is one of the following:

Option	Effect
`-a`	Displays current DS3 line alarms.
`-b on`	Transmits a DS3 Alarm Indication Signal (Blue Alarm).
`-b off`	Stops transmitting a DS3 Alarm Indication Signal (Blue Alarm).
`-c`	Displays and clears line error statistics.
`-d 1 - 7`	Displays current DS2 line state.
`-i` on	Internally loops back the DS3 payload.
`-i` off	Halt internal loop back.
`-l` on	Externally loops back the DS3 payload.
`-l` off	Halt external loop back.
`-s`	Displays line error statistics without clearing.
`-t`	Toggles debug output.
`-?`	Displays this summary.

To display alarms on the line:

```
t3-15> ds3link -a
   Loss of Signal:            false
   Out of Frame:              false
   Alarm Indication Signal:   false
   Idle Signal:               false
   Yellow Signal:             false
   In Red Alarm:              false
   C-bit parity framing:      false
```

A display of `true` for C-bit parity framing would not indicate an alarm state, but that the far end is using C-bit parity.

To display and clear line error statistics:

```
t3-1/15> ds3link -c
   Line Code Violations:    2136611
   Framing Errors:          67279
   Excessive Zeros:         2098353
   P-bit Parity Errors:     217318
   C-bit Parity Errors:     0
   Far End Block Errors:    0
   DS2 1 Framing Errors:    8415
   DS2 2 Framing Errors:    8415
   DS2 3 Framing Errors:    8415
   DS2 4 Framing Errors:    8415
   DS2 5 Framing Errors:    8415
   DS2 6 Framing Errors:    8415
   DS2 7 Framing Errors:    8415
   Statistics cleared.
```

To display the line state of the third DS2:

```
t3-1/15> ds3link -d 3
State of DS2 3:
Out of Frame:              false
Alarm Indication Signal:   false
Yellow Signal:             false
In Red Alarm:              false
Reserved Bit:              false
```

Performing an external loopback

To perform an external loopback test, use the `-l` option as follows:

```
t3-1/15> ds3link -l on
DS3 remote loopback activated

t3-1/15> ds3link -l off
DS3 remote loopback deactivated
```

Performing an internal loopback

An internal DS3 loopback connects the DS3 receive path to the DS3 transmit path at the D3. The transmitted DS3 signal is still sent to the network.

To perform an internal loopback test, use the `-i` option as follows:

```
t3-1/15> ds3link -i on
DS3 internal loopback activated

t3-1/15> ds3link -i off
DS3 internal loopback deactivated
```

Note: DS1 external loopbacks can be invoked manually with the FE-Loop command on the DS3 card. In addition, you can display DS1 error statistics with the T1-Stats command. To use these commands, first use the Open command to open a session with the card, as described at the beginning of this section.

Administering E1 and E1 FrameLine cards

The E1-Stats command reports DS1-level line errors on E1 cards. Before entering it, use the Open command to open a session with the installed card. For example, to open a session with a card in shelf 1, slot 13:

```
admin> open 1 13
```

Then enter the E1-stats command. The following example shows the command's syntax:

```
e1-1/13> e1-stats
e1-stats [ -c ] <line>  get error statistics for the line
    -c: reset statistics to zero
```

To view DS1-level statistics on the first line on the card:

```
e1-1/13> e1-stats 1
  DS1 Line 1:
CRC Errors:                  0
Frame Slips:                 9872
Framing Bit Errors:          0
Far End Block Errors:        0
Line Code Violations:        0
    Statistics cleared.
```

To view and reset the statistics to 0 (zero) on line 2:

```
e1-1/13> e1-stats -c 2
Line 2:
CRC Errors:                  0
Frame Slips:                 9872
Framing Bit Errors:          0
Far End Block Errors:        0
Line Code Violations:        0
    Statistics cleared.
```

The `Statistics cleared` message at the end of the display indicates that the statistics have been reset to 0 (zero), because the command included the `-c` option. Table 3-2 explains the E1-Stats fields.

Table 3-3. E1-Stats command fields

Field	Event that increments the field
CRC Errors	A CRC-6 checksum shows data corruption in the signal.
Frame Slips	The MAX TNT receives E1 data at a frequency higher or lower than the internal line clock. In the process of realigning itself to the transmitter, the MAX TNT can skip or repeat a frame.
Framing Bit Errors	Framing bit errors occur when the MAX TNT receives E1 data at a frequency higher or lower than that of the internal line clock. In the process of realigning itself to the transmitter, the MAX TNT can skip or repeat a frame.
Out of Frame Events	The MAX TNT no longer detects a framing pattern in the receiving signal, or it detects a pattern at a different relative offset than expected.
Line Code Violations	The MAX TNT detected either a Bipolar Violation or Excessive Zeros, which means that one of the low-level E1 rules for encoding data was violated in the received signal.
Far end block errors	The far end reported an error in an E1 frame transmitted by the MAX TNT.

Administering HDLC cards

The HDLC command displays information about HDLC channels. You can also use it to display information about the Serial Communications Adapters (SCAs) on the FrameLine cards, which are responsible for receiving and transmitting HDLC frames.

This command uses the following syntax:

```
admin> hdlc -option
```

where ***-option*** may be one of the following:

Option	Effect
-a	Displays all available HDLC channels.
-d	Displays disabled HDLC channels.
-f	Displays failed/non-existent HDLC channels.
-i	Displays in-use HDLC channels.
-p	Displays all possible HDLC channels.

This output shows the interface address of each HDLC channel in the slot, followed by the operational status, up-state, required-state, and admin-state of the channel.

For more information about the HDLC command, see the *MAX TNT Reference Guide.*

Administering ADSL cards

For ADSL-card administration, you can use MAX TNT diagnostic commands to perform BER tests and loopbacks.

Performing a ADSL BER test

When you perform a ADSL BER test, the remote end must also activate a BER test. If the remote end is not connected, an analog loopback is performed. During an analog loopback, the card itself is looped back. No remote device is involved.

Note: A BER test interrupts normal data transmission.

To perform a BER test:

1. Read in the ADSL-CAP-Statistics profile. For example, to test the ADSL line in shelf 1, slot 11, port 1:

   ```
   admin> read adsl-cap-stat {1 3 1}
   ADSL-CAP-STAT/{ shelf-1 slot-3 1 } read
   ```

2. List the contents of the Physical-Statistic profile. For example:

   ```
   admin> list physical-statistic
   [in ADSL-CAP-STAT/{ shelf-1 slot-3 1 }:physical-statistic]
   line-up-timer = { 0 0 0 }
   rx-signal-present = no
   ```

```
line-quality = 0
up-dwn-cntr = 0
self-test = passed
rs-errors = 0
rs-corrected-errors = 0
transmit-power = 0
rx-attenuation = 2
connection-sq = 0
hdlc-rx-crc-error-cnt = 0
bert-timer = 2 minutes
bert-enable = no
bert-operation-state = stopped
bert-error-counter = 0
```

3 Specify the duration of the BER test. For example:

```
admin> set bert-timer=5 minutes
```

4 Enable the BER test:

```
admin> set bert-enable=yes
```

5 Write the profile to start the BER test:

```
admin> write
```

6 Re-read the ADSL-CAP-Stat profile to update the error count parameters:

```
admin> read adsl-cap-stat
```

7 List the Physical-Statistic subprofile:

```
admin> list physical-statistic
```

The BERT-Error-Counter displays the BER test errors.

The BERT-Operation-State displays the state of the connection.

8 To end the BER test:

- If there is a remote device connected to the line, the BER test runs until the BERT-Timer expires.
- If there is no remote device connected to the line, end the test by disabling the BER test, as follows:

  ```
  admin> set bert-enable=no
  ```

9 Write the profile to end the BER test:

```
admin> write
```

Performing loopbacks

During POST, the ADSL-DMT and ADSL-CAP cards performs loopbacks on all channels. But also, you can use the XDSLcmd command to manually loop back the channels on the ADSL cards.

To use the XDSLcmd command, first open a session to the card, then enter the XDSLcmd command using the following syntax:

```
xdslcmd [-?][-l channel] [count] [bufferSize]]
```

where

- **`-l channel`** Initiates a loopback on the specified channel (from 0 to 5). If no channel is specified, all channels are tested.
- **`-?`** Displays help.
- **`count`** Specifies the number of looped frames. The default is 10.
- **`bufferSize`** Specifies the size of the looped frames. The default is 128 bytes.

The following example shows how to run a loopback test on channel 2 of an ADSL card in shelf 1, slot 6:

```
admin> open 1 6
adsl-1/6> xdslcmd -l 2
```

The loopback terminates when the **`count`** is reached.

Administering IDSL cards

The BRIchannels and BRIdisplay commands display information about IDSL cards in the MAX TNT. You can use the IDSLcmd command to perform line loopbacks and test for block errors.

Using the BRIchannels command

To show IDSL channel information, enter the BRIchannels command:

```
admin> brichannels -a|-d|-c|-i
```

where

- **`-a`** Displays all available BRI channels.
- **`-d`** Displays disabled BRI channels.
- **`-c`** Displays all possible BRI channels.
- **`-i`** Displays in-use BRI channels.

For example:

```
admin> brichannels -a

BRI channels available for use:

                                    (dvOp    dvUpSt     dvRq     sAdm  )
  Channel { {  1  9  1 }  1 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  1 }  2 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  2 }  1 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  2 }  2 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  3 }  1 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  3 }  2 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  4 }  1 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  4 }  2 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  5 }  1 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  5 }  2 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  6 }  1 }   (Up      Assigned    UP       UP    )
  Channel { {  1  9  6 }  2 }   (Up      Idle        UP       UP    )
  Channel { {  1  9  7 }  1 }   (Up      Idle        UP       UP    )
  Channel { {  1  9  7 }  2 }   (Up      Idle        UP       UP    )
```

```
  Channel { {  1  9  8 }  1 }   (Up      Idle        UP      UP    )
  Channel { {  1  9  8 }  2 }   (Up      Idle        UP      UP    )
  Channel { {  1  9  9 }  1 }   (Up      Idle        UP      UP    )
  Channel { {  1  9  9 }  2 }   (Up      Idle        UP      UP    )
  Channel { {  1  9 10 }  2 }   (Up      Idle        UP      UP    )
  Channel { {  1  9 11 }  1 }   (Up      Idle        UP      UP    )
  Channel { {  1  9 11 }  2 }   (Up      Idle        UP      UP    )
[More? <ret>=next entry, <sp>=next page, <^C>=abort]
```

Regardless of which option you enter, the BRIchannels command displays the following information:

Column Name	Description
dvOp	The operational state of the BRI channel. Values can be: • Down • Up
dvUpSt	The up status of the BRI channel. Values can be: • Idle • Reserved • Assigned
dvRq	The required state of the BRI channels. Values can be: • Down • Up
SAdm	The desired state of the device. Values can be: • Down • Up

Using the BRIdisplay command

The BRIdisplay command is a card-level command that displays actual bytes of the traffic on an IDSL card or its channels. The system monitors the traffic for the number of bytes you specify in the following syntax:

```
idsl-1/9> bridisplay count [channel]
```

where **count** is the number of bytes to display in the command's output and **channel** is a number in the range of 0-31, specifying one of the 32 D channels on the card. If you do not specify a channel, the display includes all traffic on all the card's D channels.

To use the BRIdisplay command, first open a session with the IDSL card. Then enable the debug display and specify the number of bytes and (optionally) the D channel to monitor. The following example displays up to 12 bytes of every packet on every D channel on the card:

```
admin> open 1 7
idsl-1/7> debug on
Diagnostic output enabled

idsl-1/7> bridisplay 12
Display the first 12 bytes of BRI messages
```

```
BRI-XMIT-2/2:  : 4 of 4 octets
7066CFD0: 02 81 01 09                                            ....
BRI-RECV-2/2:  : 4 of 4 octets
705FD7E0: 02 81 01 0b
..
..
..
```

To turn off the display, set `count` to zero, as in the following example:

```
idsl-1/7> bridisplay 0
```

Alternatively, you can disable debug output:

```
idsl-1/7> debug off
```

Using the IDSLcmd command

Use the IDSLcmd command to loop back the IDSL line and test for near-end block errors (NEBE) and far-end block errors (FEBE). For more information about IDSL loopbacks, see “Line loopbacks” on page 3-31. For more information about block errors, see “Block-error counters” on page 3-32.

The IDSLcmd command has the following syntax:

idslcmd -option [*channel*] [*EOC address*] [*count*] [*buffer size*]

where option is one of the following:

Option:	Effect
-1	Loops back the first B channel.
-2	Loops back the second B channel.
-a	Starts an analog loopback to test the IDSL hardware.
-f	Fetches block-error counters.
-z	Clears block-error counters.
-c	Starts sending corrupt CRCs.
-u	Cancels corrupt CRCs.
-r	Requests corrupt CRCs.
-n	Cancels the request for corrupt CRCs.
-?	Displays this summary.

The other syntax elements are:

Option:	Effect
channel	Enter the number of the channel to test (0-31). If you do not enter a channel number, the command applies to all channels.

Option:	Effect
`EOC address`	For a loopback, specifies the Embedded Operations Channel (EOC) address from which the MAX TNT rolls back the signal. The default of 0 (zero) specifies the remote ISDN TA device. A number from 1 to 6 specifies the number of an ISDN repeater between the MAX TNT and the remote TA. The repeater nearest the MAX TNT is number 1. The number 7 specifies that the EOC command should be broadcast to all the nodes on the IDSL connection.
`count`	For a loopback, specifies the number of frames to send in the loopback. The default is 10.
buffer size	For a loopback, specifies the size of the frames sent in the EOC loopback. The default is 128 bytes.

Performing IDSL diagnostics

The diagnostic functions of the IDSL line card do not use either the D or B channels to transmit diagnostic function or signaling information. These diagnostic commands are sent in the M1, M2, and M3 bits of the U-superframe. For more information about the M1, M2, and M3 bits of the superframe, refer to ANSI T1-601, from ANSI 1991.

The MAX TNT provides the following diagnostic capabilities:

- Line loopbacks
- Block error counters

Line loopbacks

During a line loopback, the MAX TNT continuously sends out test frames over the D channel to the remote end. The frames transmitted differ in content in order to cover every possible bit pattern. Note that loopback occurs at the remote end of the connection, where the remote device loops back all the frames it receives on the BRI interface.

Note: Normal data traffic is disrupted when the line is in loopback mode.

When the MAX TNT is in loopback mode:

- The MAX TNT increments the XMIT field in the loopback display each time it transmits a frame.
- When the MAX TNT receives a frame that matches the transmitted frame in both size and the content, it sends out a new frame and increments the RECV field in the loopback display.
- When the MAX TNT receives a frame that does not match the transmitted frame, it still sends out a new frame, but does not increment the receive counter.
- If the MAX TNT does not receive back a frame within the timeout period (4 seconds) it sends out another frame without incrementing the RECV counter.

Note that you cannot request that the remote end stop sending corrupt CRCs while the MAX TNT is in loopback mode.

To put the line into loopback mode:

1 Open a session with the IDSL card. For example, to connect to an IDSL card on shelf 1, slot 4:

```
admin> open 1 4
idsl-1/4>
```

2 Start the loopback. For example, to loop back the first B channel on the first IDSL line:

```
idsl-1/4>idslcmd -1 1

IDSL-0/1: EOC loopback on B1 channel EOC address:0 count:10 size:128
MUNICH-0/1: timeout TEST
MUNICH-0/1 received 0 of 10 with 0 errors
IDSL-0/1: failed TEST
```

The numbers displayed are cumulative totals, starting when the Line loopback command is issued. Each time the loopback command is started or restarted, the loopback counters are reset to 0.

Note: Only one loopback can be issued at a time on the same line.

Block-error counters

The block-errors counters displays the far-end block errors (FEBE) and near-end block errors (NEBE) that have occurred since the error buffers were cleared. The totals for each buffer return to zero after they reach 65535. The block-error totals are obtained from the remote device.

A near-end block error (NEBE) indicates that the MAX TNT has detected an error in a packet it has received. A far-end block error (FEBE) indicates that the remote device has detected an error in a packet it has received. In either case, the block errors shown in this display are stored in the far-end device's registers.

You can test the NEBE and FEBE counters by simulating transmission errors with artificially corrupted CRCs.

You can use the block error counters to monitor transmission quality at the U-interface. A block error is detected each time the calculated checksum of the received data does not correspond to the control checksum transmitted in the successive superframe. One block error indicates that one U-superframe has not been transmitted correctly. The block error count does not provide information regarding the number of bit errors in the U-superframe, but only indicates that the CRC failed in that superframe.

Testing the far-end block error counters

To test the FEBE counters:

1 Open a session with the IDSL card. For example, to connect to an IDSL card on shelf 1, slot 4:

```
admin> open 1 4
idsl-1/4>
```

2 Send corrupt CRCs over the B channel. For example, to test the first IDSL line's FEBE counter:

```
idsl-1/4>idslcmd -c 1
IDSL: corrupt CRC for channel:1
```

3 Display the FEBE counter. For example, to display the FEBE for channel 1:

```
idsl-1/4>idslcmd -f 1
     FEBE    NEBE
1:   0       0
```

4 To cancel the transmission of corrupt CRCs, use the `-u` option. For example:

```
idsl-1/4>idslcmd -u 1
IDSL: cancel corrupt CRC for channel:1
```

Testing the near-end block error counters

To test the NEBE counters:

1 Open a session with the IDSL card. For example, to connect to an IDSL card on shelf 1, slot 4:

```
admin> open 1 4
idsl-1/4>
```

2 Request corrupt CRCs over the B channel. For example, to test the first IDSL line's NEBE counter:

```
admin>idslcmd -r 1
IDSL: request corrupt CRC for channel:1
```

3 Display the FEBE counter. For example, to display the NEBE for channel 1:

```
admin>idslcmd -f 1
     FEBE    NEBE
 1:   0       0
```

4 To cancel the reception of corrupt CRCs, use the `-n` option. For example:

```
admin>idslcmd -n 1
IDSL: cancel corrupt CRC for channel:1
```

Clearing the error registers

To clear the NEBE registers, use the `-z` option. For example:

```
admin>idslcmd -z
IDSL: clear block error counters for channel:1
```

The error register totals are also reset when the totals reach 65535.

Administering SDSL cards

You can use the MAX TNT SDSL diagnostic commands to display information about SDSL channels and to initiate loopbacks. You can also obtain SDSL information with the Read and List commands.

Using the SDSLlines command

To show SDSL channel information, enter the SDSLlines command:

```
admin> sdsllines -a|-d|-f|-u
```

where

- `-a` Displays all available channels.
- `-d` Displays the disabled channels.
- `-f` Displays all possible channels.
- `-u` Displays in-use channels.

Example: sdsllines -a

```
All SDSL lines:
                                (dvOp   dvUpSt     dvRq    sAdm    nailg)
  Line    {    1  3  1 }     (Up      Idle      UP      UP     00001)
  Line    {    1  3  2 }     (Up      Assigned  UP      UP     00002)
  Line    {    1  3  3 }     (Up      Assigned  UP      UP     00003)
  Line    {    1  3  4 }     (Up      Idle      UP      UP     00004)
  Line    {    1  3  5 }     (Up      Idle      UP      UP     00005)
  Line    {    1  3  6 }     (Up      Assigned  UP      UP     00006)
  Line    {    1  3  7 }     (Up      Idle      UP      UP     00007)
  Line    {    1  3  8 }     (Up      Assigned  UP      UP     00008)
  Line    {    1  3  9 }     (Up      Assigned  UP      UP     00009)
  Line    {    1  3 10 }     (Up      Assigned  UP      UP     00010)
  Line    {    1  3 11 }     (Up      Assigned  UP      UP     00011)
  Line    {    1  3 12 }     (Up      Assigned  UP      UP     00012)
  Line    {    1  3 13 }     (Up      Assigned  UP      UP     00013)
  Line    {    1  3 14 }     (Up      Assigned  UP      UP     00014)
  Line    {    1  3 15 }     (Up      Assigned  UP      UP     00015)
  Line    {    1  3 16 }     (Up      Idle      UP      UP     00016)
```

In addition to identifying SDSL channels, SDSLlines output includes the following fields:

Column Name	Description
`dvOp`	The operational state of the SDSL channel. Values can be: • Down • Up • None

Column Name	Description
dvUpSt	The up status of the SDSL channel. Values can be: • Idle • Reserved • Assigned
dvRq	The required state of the SDSL channels. Values can be: • Down • Up
SAdm	The desired state of the device. Values can be: • Down • Up
nailg	The nailed group that this line is assigned to.

Using the XDSLcmd command

During POST, the SDSL card performs a loopback on all channels. But also, you can use the XDSLcmd command to manually loop back the channels on the SDSL card.

The XDSLcmd command uses the following syntax:

```
xdslcmd [-?][-l channel] [count] [bufferSize]]
```

where

- `-l channel` Initiates a loopback on the specified channel (from 0 to 15). If no channel is specified, all channels are tested.
- **`-?`** Displays help.
- **`count`** Specifies the number of looped frames. The default is 10.
- **`bufferSize`** Specifies the size of the looped frames. The default is 128 bytes.

The following example shows how to run a loopback test on channel 8 of a SDSL card in shelf 1, slot 6:

```
admin> open 1 6
sdsl-1/6> xdslcmd -l 8
```

The loopback terminates when the **`count`** is reached.

Troubleshooting SDSL connections

If the SDSL link between the MAX TNT and the remote end does not come up after a few seconds, try the troubleshooting steps described in Table 3-4.

Table 3-4. troubleshooting SDSL connections

Action	Example
Verify that the card is active.	`admin>` **`read sdsl-stat {1 1 1}`** `admin>` **`list`** Check the list to verify that `active=yes`.
Verify that the card passed POST.	`admin>` **`read sdsl-stat {1 1 1}`** `admin>` **`list physical-statistic`** Check the list to verify that `self test=passed`.

Administering SWAN cards

To show serial WAN line information, enter the SWANlines command:

```
admin> swanlines -a|-d|-f|-u
```

where

- `-a` Displays all available channels.
- `-d` Displays the disabled channels.
- `-f` Displays all free channels.
- `-u` Displays in-use channels.

Although SWAN cards do not yet support compression, additional POST tests have been added to test the compression hardware on the cards. If a SWAN card fails POST after you upgrade to 2.0.0 or above, the new tests have probably detected a problem with the compression chip, which requires that you return the card.

Administering UDS3 cards

The UDS3lines and UDS3dump commands enable you to monitor the UDS3 card.

Using the UDS3lines command

This command uses the following syntax:

```
admin> uds3lines -option
```

where ***-option*** may be one of the following:

Option	Effect
-a	Displays all available UDS3 lines.
-d	Displays disabled UDS3 lines.
-f	Displays free UDS3 lines.
-u	Displays in-use UDS3 lines.

In the following example, the UDS3lines command displays the all UDS3 lines:

```
admin> uds3lines -a
 All UDS3 lines:
                                (dvOp   dvUpSt  dvRq    sAdm    nailg)
  Line    {    1 13  1 }       (Up      Idle    UP      UP     00131)
```

Regardless of which option you enter, the UDS3lines command displays the following information:

Column Name	Description
dvOp	The operational state of the UDS3 line. Values can be: • Down • Up
dvUpSt	The up status of the UDS3 line. Values can be: • Idle • Reserved • Assigned
dvRq	The required state of the UDS3 line. Values can be: • Down • Up
SAdm	The desired state of the device. Values can be: • Down • Up
nailg	The nailed group that this line is assigned to.

Using the UDS3Dump command

The UDS3dump card-level command displays the information about the DS3 interface. To use this command, first open a session to the UDS3 card, then issue the UDS3dump command, using the following syntax:

```
uds3-1/11> uds3dump interval
```

where ***interval*** may be one of the following:

Option	Effect
0	Displays the DS3 MIB (RFC 1407) dsx3CurrentTable.
1-96	Displays the DS3 MIB (RFC 1407) dsx3IntervalTable.
97	Displays the DS3 MIB (RFC 1407) dsx3TotalTable.

In the following example, the UDS3dump command displays the current interval table:

```
uds3-1/13> uds3dump 0
Index PESs  PSESs SEFSs UASs  LCVs  PCVs  LESs  CCVs  CESs CSESs
-------------------------------------------------------------
0     0     0     0     1     0     0     0     0     0     0
```

The output contains the following fields (Refer to RFC 1407 for complete description of these errors.):

Field	Description
PESs	A P-bit errored second is a second during which one of the following error conditions occurs: • A P-Bit error • An out of frame error • An incoming A1S signal Note that the count is not incremented by the number of unavailable seconds.
PSESs	A P-bit severely errored second is a second during which one of the following error conditions occurs: • There are 44 or more P-Bit errors • An out of frame error • An incoming A1S signal Note that the count is not incremented by the number of unavailable seconds.
SEFSs	A severely errored framing second is a second during which one of the following error conditions occurs: • An out of frame error • An incoming A1S signal
UASs	The number of seconds the interface is unavailable. Note that only LES and SEFS errors are counted while the interface is unavailable.
LCVs	A line coding violation error is the sum of bipolar (BPV) and excessive zero (EXZ) errors. An excessive zero error increments the count by one no matter how many zeros are transmitted.
PCVs	P-bit errors indicate that MAX TNT received a P-bit code on the DS3 M-frame that differs from the locally calculated code.

Field	Description
LESs	A line errored seconds is a second during which one of the following error conditions occurs: • A C-bit coding violation error • A loss of signal error
CCVs	A C-bit coding violation error indicates a parity error.
CESs	A C-bit errored second is a second during which one of the following error conditions occurs: • A C-bit coding violation error • An out of frame error • An incoming A1S signal This applies only to SYNTRAN and C-bit Parity DS3 lines. Note that the count is not incremented by the number of unavailable seconds.
CSESs	A C-bit severely errored second is a second during which one of the following error conditions occurs: • There are 44 or more C-bit coding violation errors • An out of frame error • An incoming A1S signal This applies only to SYNTRAN and C-bit Parity DS3 lines. Note that the count is not incremented by the number of unavailable seconds.

Administering modems

The MAX TNT provides diagnostic commands to display modem status, bring modems or channels up or down, or quiesce modems.

Using the Modem command to display modem status

To show modem information, enter the Modem command:

```
modem -a|-d|-f|-g|-i|-m|-s
```

where

- `-a` Displays all available modems.
- `-d` Displays the disabled channels.
- `-f` Displays failed or non-existent modems.
- `-g` Displays available good modems.
- `-i` Displays in-use modems.
- `-m` Displays all possible modems.
- `-s` Displays suspect modems.

For example, to see which modems are in use:

```
admin> modem -i
Modems allocated/in-use
                              (dv0p dvUpSt DvRq sAdm)
Modem {1 14 1}                (Up   Assign UP   UP )
```

For more information about the Modem command refer to the *MAX TNT Reference Guide*.

Bringing a modem or channel up or down

To administratively up or down a device, you can use the Device command or a Device-State profile. (For discussion of Device-State profiles, see “Using the Device-State profile” on page 9-6.)

For example, to administratively down modem 24 in slot 3 on shelf 1:

```
admin> device -d {{1 3 24} 0}
```

To bring the modem back up:

```
admin> device -u {{1 3 24} 0}
```

Disabling a modem

To disable a modem:

1. Read in the LAN Modem profile. For example:

   ```
   admin> read LAN-Modem
   LAN-MODEM/{ shelf-1 slot-2 0 } read
   ```

2. Disable the modem:

   ```
   admin> set modem-disable-mode 1= disable
   ```

3. Write the profile to commit your changes:

   ```
   admin> write
   LAN-MODEM/{ shelf-1 slot-2 0 } written
   ```

Quiescing digital modems

The system creates a LAN-Modem profile for each installed modem card. Removing or downing a modem card does not delete this profile or change its contents. You can use the LAN-Modem profile to quiesce digital modems. Quiescing a modem makes it available for maintenance in a graceful way, not by tearing down the current connection, but by taking the channel out of service as soon as the connection is dropped.

To use a LAN-Modem profile, first open it and list its contents. For example:

```
admin> read lan {1 6 0}
LAN-MODEM/{ shelf-1 slot-6 0 } read

admin> list
physical-address* = { shelf-1 slot-6 0 }
modem-disable-mode = [ enable enable enable enable enable +
```

Then, to quiesce a modem, list its Modem-Disable-Mode setting and change it to `disable`. For example:

```
admin> list modem-dis
...(All 48 modem settings are displayed)

admin> list 20
admin> set = disable
admin> write
LAN-MODEM/{ shelf-1 slot-6 0 } written
```

To bring the modem back up:

```
admin> set = enable

admin> write
LAN-MODEM/{ shelf-1 slot-6 0 } written
```

Note: When you quiesce a modem, you can also quiesce an arbitrary idle T1 channel at the same time by using the Dis-Channel setting. For details, see the *MAX TNT Reference Guide*.

Network Administration

4

This chapter covers the following topics:

The MAX TNT supports several network management commands, which are useful for locating the sources of problems on the network and for communicating with other hosts for management purposes.

Some of the network management tools focus on routing and interface information. They enable you to display the routing and interface tables, view real-time routing statistics, display route caches, and make changes to the routing table. The OSPF command supports numerous arguments for viewing information about the OSPF link-state database, adjacencies, and other aspects of the router configuration.

Other tools are geared toward network usage, and enable you to display packets received on LAN interfaces, display the ARP cache, Ping a host, and log into a host by means of Rlogin or Telnet.

For complete information about the commands described in this chapter, see the *MAX TNT Reference Guide*.

Diagnostic tools for TCP/IP networks

The MAX TNT maintains an internal IP routing table. You can configure the system to use RIP or OSPF to propagate the information in that table to other routers, receive information from other routers, or both, on any LAN or WAN interface. For information about configuring the router, see the *MAX TNT Network Configuration Guide*.

Using the Ping command to test connectivity

The Ping command is useful for verifying that the transmission path between the MAX TNT and another station is open. Ping sends an ICMP echo_request packet to the specified station. It the station receives the packet, it returns an ICMP echo_response packet. For example, to Ping the host `techpubs`:

```
admin> ping techpubs
```

```
PING techpubs (10.65.212.19): 56 data bytes
64 bytes from 10.65.212.19: icmp_seq=0 ttl=255 time=0 ms
64 bytes from 10.65.212.19: icmp_seq=3 ttl=255 time=0 ms
^C
--- techpubs ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
```

You can terminate the Ping exchange at any time by pressing Ctrl-C. When you press Ctrl-C, the command reports the number of packets sent and received, the percentage of packet loss, the number of duplicate or damaged echo_response packets (if any), and round-trip statistics. In some cases, round-trip times cannot be calculated.

During the Ping exchange, the MAX TNT displays information about the packet exchange, including the Time-To-Live (TTL) of each ICMP echo_response packet.

The maximum TTL for ICMP Ping is 255, while and the maximum TTL for TCP is often 60 or lower, so you might be able to Ping a host but not be able to run a TCP application (such as Telnet or FTP) to that station. If you Ping a host running a version of Berkeley UNIX before 4.3BSD-Tahoe, the TTL report is 255 minus the number of routers in the round-trip path. If you Ping a host running the current version of Berkeley UNIX, the TTL report is 255 minus the number of routers in the path from the remote system to the station performing the Ping.

Using the Netstat command to display the interface table

At system startup, the MAX TNT creates an IP interface, in the active state, for each Ethernet interface that has a configured IP-Interface profile, and for the built-in loopback, reject, and blackhole interfaces. It also creates IP interfaces in the inactive state for remote connections. For each IP interface that is not configured as a private route, the MAX TNT also adds a route to the routing table.

IP interfaces change between the active and inactive state as switched calls are brought up and down. To display the interface table, enter the Netstat command with the `-in` option, as in the following example:

```
admin>netstat -i
Name        MTU   Net/Dest          Address         Ipkts    Ierr Opkts    Oerr
ie0          1500 192.168.7.0/24    192.168.7.135     71186    2    53131   96
lo0          1500 127.0.0.1/32      127.0.0.1         53195    0    53195    0
rj0          1500 127.0.0.2/32      127.0.0.2             0    0        0    0
bh0          1500 127.0.0.3/32      127.0.0.3             0    0        0    0
wanabe       1500 127.0.0.3/32      127.0.0.3             0    0        0    0
local       65535 127.0.0.1/32      127.0.0.1         59753    0    59753    0
mcast       65535 224.0.0.0/4       224.0.0.0             0    0        0    0
tunnel7      1500 192.168.7.0/24    192.168.7.135         0    0        0    0
vr0_main     1500 192.168.7.135/32  192.168.7.135         0    0        0    0
sip0        65535 -                 -                     0    0        0    0
wan10        1528 200.4.2.2         192.168.7.135         0    0        0    0
wan11        1528 200.5.2.2         192.168.7.135         0    0        0    0
wan12        1528 200.6.1.2         192.168.7.135         0    0        0    0
wan13        1528 200.6.2.2         192.168.7.135         0    0        0    0
wan14        1528 200.100.2.2       192.168.7.135         0    0        0    0
```

```
wan15       1528 200.100.3.2       192.168.7.135          0    0        0    0
wan16       1528 200.4.4.2         192.168.7.135          0    0        0    0
wan17       1500 200.6.100.2       200.1.100.2            0    0        0    0
wan18       1528 200.4.4.3         192.168.7.135          0    0        0    0
wan19       1528 200.4.2.3         192.168.7.135          0    0        0    0
wan20       1528 200.3.2.2         192.168.7.135          0    0        0    0
wan21       1528 200.3.1.2         192.168.7.135          0    0        0    0
wan22       1528 200.4.103.2       192.168.7.135          0    0        0    0
wan23       1500 200.4.101.3       200.2.101.2            0    0        0    0
..
..
..
ie1-5-1     1500 200.1.1.0/24      200.1.1.2              0    0        1    0
ie1-5-2     1500 200.1.2.0/24      200.1.2.2              0    0        1    0
ie1-5-3     1500 200.2.1.0/24      200.2.1.2          75837    0    75838    0
ie1-5-4     1500 200.2.2.0/24      200.2.2.2              0    0        1    0
ie1-5-5     1500 -                 -                      0    0        0    0
```

The interface table contains the following information:

Column name	Description
`Name`	Name of the interface: • `ie0-`*n* —The shelf-controller Ethernet interfaces. • `ie[shelf]-[slot]-[item]`—The Ethernet interfaces for Ethernet cards. • `lo0`—The loopback interface. • `rj0`—The reject interface, used in network summarization. • `bh0`—The blackhole interface, used in network summarization. • `wanN` —A WAN connection, entered as it becomes active. • `wanabe`—An inactive RADIUS dialout profile. • `local`—The local machine. • `mcast`—The multicast interface, which represents the multicast forwarder for the entire class-D address space. • `tunnel`*N*—A pseudo-interface that is used only when the MAX TNT is configured as an ATMP Router Home Agent. In that configuration, the MAX TNT creates a route for each registered Mobile Client. Regardless of how many tunnels the Home Agent may terminate, there is always a single tunnel interface. (The number appended to the tunnel interface name is an internal number used by the system.)
`MTU`	(Maximum Transmission Unit) The maximum packet size allowed on the interface.
`Net/Dest`	Network or the target host this interface can reach.
`Address`	Address of this interface.
`Ipkts`	Number of packets received.

Column name	Description
`Ierr`	Number of packets that contain errors.
`Opkts`	Number of packets transmitted.
`Oerr`	Number of transmitted packets that contain errors.

Displaying and modifying IP routes

This section explains how to display the MAX TNT IP routing table. It also explains how to use the Netstat command to display the IP routing table and the IProute command to add or delete static routes. For complete information about configuring IP routing on the MAX TNT, see the *MAX TNT Network Configuration Guide*.

Using the Netstat command to display the routing table

To display the routing table, enter the Netstat command with the `-r` argument, as in the following example:

```
admin>netstat -r
Destination        Gateway         IF          Flg   Pref Met     Use        Age
127.0.0.0/8        -               bh0         CP       0   0       0     154417
127.0.0.1/32       -               local       CP       0   0       0     154417
127.0.0.2/32       -               rj0         CP       0   0       0     154417
182.21.33.0/24     192.168.7.1     ie0         SG      60   8       0     150873
192.168.7.0/24     -               ie0         C        0   0   50041     154417
192.168.7.135/32   -               local       CP       0   0    2522     154417
..
..
..
216.64.222.0/24    192.168.7.1     ie0         SG      60   8    1456     150873
224.0.0.0/4        -               mcast       CP       0   0       0     154417
224.0.0.1/32       -               local       CP       0   0       0     154417
224.0.0.2/32       -               local       CP       0   0       0     154417
224.0.0.5/32       -               local       CP       0   0       0     154417
224.0.0.6/32       -               local       CP       0   0       0     154417
224.0.0.9/32       -               local       CP       0   0       0     154417
255.255.255.255/32 -               ie0         CP       0   0       0     154417
```

The columns in the routing table contain the following information:

Column	Description
`Destination`	The route's target address. To send a packet to this address, the MAX TNT uses this route. If the target address appears more than once in the routing table, the MAX TNT uses the most specific route (having the largest subnet mask) that matches that address.
`Gateway`	The next hop router that can forward packets to the given destination. Direct routes (without a gateway) show a hyphen in this column.

Column	Description
`IF`	The name of the interface through which to send packets over this route: • `ie0` or `ie[`*`shelf`*`]-[`*`slot`*`]-[`*`item`*`]` is an Ethernet interface. • `lo0` is the loopback interface. • `rj0` is the reject interface, used in network summarization. • `bh0` is the blackhole interface, used in network summarization. • `wan`*`N`* is a WAN connection, entered as it becomes active. • `wanabe` indicates an inactive RADIUS dialout profile. • `local` indicates a single route targeted at the local machine. • `mcast` indicates a route to a virtual device. The route encapsulates the multicast forwarder for the entire class D address space.
`Flg`	One or more of the following flags: • `C`—a directly connected route, such as Ethernet • `I`—an ICMP redirect dynamic route • `N`—placed in the table via SNMP MIB II • `O`—A route learned from OSPF • `R`—a route learned from RIP • `r`—a transient RADIUS-like route • `S` —a static route • `?`—a route of unknown origin, which indicates an error • `G`—an indirect route via a gateway • `P`—a private route • `T`—a temporary route • `M`—a multipath route • `*`—a backup static route for a transient RADIUS-like route
`Pref`	The preference value. See the description of the Preference parameter for information about defaults for route preferences.
`Metric`	A RIP-style metric for the route, with a range of 0-16. Routes learned from OSPF show a RIP metric of 10. OSPF cost-infinity routes show a RIP metric of 16.
`Use`	A count of the number of times the route was referenced since it was created. (Many of these references are internal, so this is not a count of the number of packets sent over this route.)
`Age`	The age of the route in seconds. RIP and ICMP entries are aged once every 10 seconds.

Modifying the routing table

The IProute command enables you to manually add routes to the routing table, delete them, or change their preference or metric values. The command is useful for temporary routing changes. Changes you make to the routing table with the IProute command do not persist across system resets. RIP and OSPF updates can add back any route you remove with IProute Delete. Also, the MAX TNT restores all routes listed in the IP-Route profile after a system reset.

The IProute command uses the following syntax:

iproute option

Syntax element	Description
add	Add an IP route to the routing table.
delete	Delete an IP route from the routing table.

Adding a static IP route to the routing table

To add a static IP route to the MAX TNT unit's routing table, use the IProute Add command:

iproute add *dest_IPaddr* [**/*subnet_mask***] ***gateway_IPaddr*** [**/*subnet_mask***] [**pref**] [**metric**]

Syntax element	Description
dest_IPaddr [**/subnet_mask**]	Destination network address. The optional subnet mask specifies the number of bits in the mask. The default is 0.0.0.0/0. Note that the router uses the most specific route (having the largest mask) that matches a given destination.
gateway_IPaddr [**/subnet_mask**]	IP address of the router that can forward packets to the destination network, and optional subnet mask (in bits). The default is 0.0.0.0.
pref	Route preference. The default is 100.
metric	Virtual hop count of the route. You can enter a value between 1 and 15. The default is 1. Note that RIP and OSPF updates can change the metric for any route, including one you have modified manually by using the IProute command.

For example, consider the following command:

```
admin> iproute add 10.1.2.0/24 10.0.0.3/24 1
```

It adds a route to the 10.1.2.0 network and all of its subnets, through the IP router located at 10.0.0.3/24. The metric to the route is 1 (one hop away).

If you try to add a route to a destination that is already in the routing table, the MAX TNT does not replace the existing route unless it has a higher metric than the route you attempt to add. If you get the message `Warning: a better route appears to exist`, the MAX TNT has rejected your attempt to add a route.

Deleting a static IP route from the routing table

To remove a static IP route from the MAX TNT unit's routing table, enter the IProute Delete command:

iproute delete
dest_IPaddr[**/*subnet_mask***][***gateway_IPaddr***[**/*subnet_mask***]]

The arguments are the same as for IP Route Add. For example, the following command removes the route to the 10.1.2.0 network:

```
admin> iproute delete 10.1.2.0 10.0.0.3/24
```

You can also change the metric or preference value of an existing route by using the IProute command. For example, if the routing table contains the following route:

```
Destination        Gateway        IF       Flg   Pref Met     Use       Age
10.122.99.0/24     10.122.99.1    wan4     SG     100   7       0     48630
```

You could change the metric as follows:

```
admin> iproute add 10.122.99.0/24 10.122.99.1 50 3
```

Using the TraceRoute command to trace routes

The TraceRoute command is useful for locating slow routers or diagnosing IP routing problems. It traces the route an IP packet follows, by launching UDP probe packets with a low Time-To-Live (TTL) value and then listening for an ICMP `time exceeded` reply from a router. For example, to trace the route to the host `techpubs`:

```
admin> traceroute techpubs

traceroute to techpubs (10.65.212.19), 30 hops max, 0 byte packets
1  techpubs.eng.ascend.com (10.65.212.19)  0 ms  0 ms  0 ms
```

Probes start with a TTL of one and increase by one until of the following conditions occurs:

- The MAX TNT receives an ICMP `port unreachable` message. (The UDP port in the probe packets is set to an unlikely value, such as 33434, because the target host is not intended to process the packets. A `port unreachable` message indicates that the packets reached the target host and were rejected.)
- The TTL value reaches the maximum value. (By default, the maximum TTL is set to 30.) You can use the `-m` option to specify a different TTL. For example:

```
admin> traceroute -m 60 techpubs

traceroute to techpubs (10.65.212.19), 60 hops max, 0 byte packets
1  techpubs.eng.abc.com (10.65.212.19)  0 ms  0 ms  0 ms
```

TraceRoute sends three probes at each TTL setting. The second line of output shows the address of the router and the round trip time of each probe. If the probe answers come from different gateways, the address of each responding system is shown. If there is no response within a three-second timeout interval, the second line of output an asterisk.

For the details of the TraceRoute command, see the *MAX TNT Reference Guide*.

Using the NSlookup command to verify name service setup

You can retrieve a host address by using the NSlookup command, provided that the MAX TNT has been configured with the address of a name server. (For information about configuring name servers, see the *MAX TNT Network Configuration Guide*). If a host has several IP interfaces, the command returns several addresses.

To retrieve the IP address of the host `techpubs`, proceed as in the following example:

```
admin> nslookup techpubs

Resolving host techpubs.
IP address for host techpubs is 10.65.212.19.
```

Using the ARPtable command to display the ARP cache

The Address Resolution Protocol (ARP) translates between IP addresses and media access control (MAC) addresses as defined in RFC 826. Hosts broadcast an ARP request that is received by all hosts on the local network, and the one host that recognizes its own IP address sends an ARP response with its MAC address.

The MAX TNT maintains a cache of known IP addresses and host MAC, addresses which enables it to act as a proxy for ARP requests for target hosts across the WAN, provided that proxy mode is turned on. (For configuring proxy ARP, see *MAX TNT Network Configuration Guide*.)

With the ARPtable command, you can display the ARP table, add or delete ARP table entries, or clear the ARP cache entirely. To display the ARP cache, enter the ARPtable command without any arguments, as in the following example:

```
admin> arptable
IP Address       MAC Address        Type  IF   Retries/Pkts/RefCnt  Time Stamp

10.103.0.141    00:B0:24:BE:D4:84   DYN   0        0/0/1            23323
10.103.0.2      00:C0:7B:7A:AC:54   DYN   0        0/0/599          23351
10.103.0.220    00:C0:7B:71:83:02   DYN   0        0/0/2843         23301
10.103.0.1      08:00:30:7B:24:27   DYN   0        0/0/4406         23352
10.103.0.8      00:00:0C:06:B3:A2   DYN   0        0/0/6640         23599
10.103.0.7      00:00:0C:56:57:4C   DYN   0        0/0/6690         23676
10.103.0.49     00:B0:80:89:19:95   DYN   0        0/0/398          23674
```

The ARP table displays the following information:

Column	Description
`IP Address`	The address contained in ARP requests.
`MAC Address`	The MAC address of the host.
`Type`	How the address was learned, that is, dynamically (DYN) or by specification of a static route (STAT).
`IF`	The interface on which the MAX TNT received the ARP request.
`Retries`	The number of retries needed to refresh the entry after it timed out.
`Pkts`	The number of packets sent out to refresh the entry after it timed out.

To add an ARP table entry, use the `-a` option, as in the following example:

```
admin> arptable -a 10.65.212.3 00:00:81:3D:F0:48
```

To delete an ARP table entry, use the `-d` option, as in the following example:

```
admin> arptable -a 10.9.8.20
```

To clear the entire ARP table, use the `-f` option:

```
admin> arptable -f
```

Displaying protocol statistics

The Netstat command displays the MAX TNT IP interface and routing tables, protocol statistics, and active sockets. By default (without an argument), the Netstat command reports information about both UDP and TCP. Following is an example that shows the use of Netstat without any arguments to display UDP and TCP socket information:

```
admin> netstat
udp:
-Socket-  Local Port  InQLen  InQMax    InQDrops    Total Rx
1/c    0        1023       0       1           0           0
1/c    1         520       0       0           0       15510
1/c    2           7       0      32           0           0
1/c    3         123       0      32           0           0
1/c    4        5150       0     256           0           0
1/c    5        1022       0     128           0           0
1/c    6         161       0      32           0           0
1/c    7        1797       0     128           0          22
1/8    0        1018       0     128           0           0
1/8    1       20108       0      32           0           0
1/8    2        1008       0     128           0           0
1/8    3        1798       0     128           0           0
1/9    0        1021       0     128           0           0
1/9    1       20109       0      32           0           0
1/9    2        1009       0     128           0           0
1/9    3        1799       0     128           0           0
1/10   0        1020       0     128           0           0
1/10   1       20110       0      32           0           0
1/10   2        1010       0     128           0           0
1/10   3        1800       0     128           0           0
1/11   0        1017       0     128           0           0
1/11   1       20111       0      32           0           0
1/11   2        1011       0     128           0           0
1/11   3        1801       0     128           0           0
1/12   0        1019       0     128           0           0
1/12   1       20112       0      32           0           0
1/12   2        1012       0     128           0           0
1/12   3        1802       0     128           0           0
```

```
tcp:
-Socket- Local                    Remote                   State
1/c    0 192.168.7.135.79         *.*                      LISTEN
1/c    1 192.168.7.135.1723       *.*                      LISTEN
1/c    2 192.168.7.135.23         *.*                      LISTEN
1/c    4 192.168.7.135.23         172.20.32.137.42863      ESTABLISHED
1/c    9 192.168.7.135.23         206.65.212.10.1991       ESTABLISHED
```

The output shows the queue depth of various UDP ports, as well as the total packets received and total packets dropped on each port. The total-packets-received count includes the total packets dropped. For this sample output, the SNMP queue depth was set to 32. For information about queue depths, see the *MAX TNT Network Configuration Guide*.

The Netstat command supports the `-s` option, which displays protocol statistics. The `-s` option uses the following syntax:

`netstat -s identifiers`

If no identifiers follow the `-s` option, all protocol statistics are shown. If specified, the identifiers determine the type of protocol statistics to display. Valid identifiers include `udp`, `tcp`, `icmp`, `ip`, `igmp`, or `mcast`. Following is an example that displays all statistics:

```
admin>netstat -s

udp:
        15636 packets received
        0 packets received with no ports
        0 packets received with errors
        0 packets dropped
        68 packets transmitted

tcp:

        0 active opens
        7 passive opens
        0 connect attempts failed
        0 connections were reset
        2 connections currently established
        1457 segments received
        0 segments received out of order
        1728 segments transmitted
        18 segments retransmitted
        5 active closes
        0 passive closes
        0 disconnects while awaiting retransmission

icmp:

        216 packets received
        0 packets received with errors
        Input histogram:
                216 echo requests


        271 packets transmitted
        0 packets not transmitted due to lack of resources
        Output histogram:
                216 echo replies
```

```
                24 destination unreachable
                31 time exceeded


ip:

        28860 packets received
        0 packets received with header errors
        0 packets received with address errors
        0 packets received forwarded
        0 packets received with unknown protocols
        0 inbound packets discarded
        17310 packets delivered to upper layers
        2084 transmit requests
        0 discarded transmit packets
        49 outbound packets with no route
        0 reassemblies timeout
        268 reassemblies required
        12 reassemblies succeeded
        244 reassemblies failed
        12 fragmentation succeeded
        0 fragmentation failed
        24 fragmented packets created
        0 route discards due to lack of memory
        64 default ttl

igmp:

        0 packets received
        0 bad checksum packets received
        0 bad version packets received
        0 query packets received
        0 leave packets received
        0 packets transmitted
        0 query packets sent
        0 response packets sent
        0 leave packets sent

mcast:

        0 packets received
        0 packets forwarded
        0 packets in error
        0 packets dropped
        0 packets transmitted
```

Logging into a network host

The Rlogin and Telnet commands enable you to log into a network host from the MAX TNT.

Using the Rlogin command

The Rlogin command initiates a login session from a host card, such as a modem or HDLC card, to a remote host. For example, to log into the host `techpubs`, first open a session with the host card. Then issue the Rlogin command:

```
hdlc-1/16> rlogin techpubs
```

```
Password:
Last login: Wed Oct  2 10:31:36 from marcel.marceau
SunOS Release 4.1.4 (TECHPUBS-BQE) #1: Wed Feb 4 08:56:59
PDT 1998
techpubs%
```

You can log out of the remote host by entering the Rlogin escape sequence (tilde-dot):

```
techpubs% ~.
Connection closed.
```

Or, you can log out explicitly:

```
techpubs% logout
Connection closed.
```

If you wish, you can change the default escape character from a tilde to any other character. For details, see the *MAX TNT Reference Guide*.

If your user name on the MAX TNT is different from your user name on the remote host, you can specify a user name on the Rlogin command line. For example:

```
admin> rlogin -l marcel techpubs
Password:
```

Using the Telnet command

The Telnet command initiates a login session to a remote host. For example, to Telnet into the host `techpubs`:

```
admin> telnet techpubs

Connecting to techpubs (10.65.212.19) ...
Escape character is '^]'
Connected
SunOS UNIX (techpubs)
```

You can close the Telnet session by logging out of the remote host:

```
techpubs% logout
Connection closed.
```

Diagnostic tools for IGMP multicast interfaces

The IGMP command displays information about IGMP groups and clients. This can be useful for tracking the IGMP group memberships and active client interfaces.

Displaying IGMP group information

To display active multicast group addresses and clients (interfaces) registered for each group, enter the IGMP command with the `group` option:

```
admin> igmp group
IGMP Group address Routing Table Up Time: 0:0:22:17
 Hash      Group Address    Members    Expire time   Counts
  10       224.0.2.250
```

```
2           0:3:24       3211 :: 0 S5
1           0:3:21       145 :: 0 S5
0(Mbone)    ......       31901 :: 0 S5
```

The output contains the following fields:

Field	Description
`Hash`	Index to a hash table (displayed for debugging purposes only).
`Group address`	IP multicast address used for the group. An asterisk indicates the IP multicast address being monitored, meaning that members join this address by local application.
`Members`	ID of each member of each multicast group. The zero ID represents members on the same Ethernet interface as the MAX TNT. All other IDs go to members of each group as they inform the MAX TNT that they have joined the group. If a client is a member of more than one group to which the MAX TNT forwards multicast packets, it has more than one multicast ID.
`Expire time`	When this membership expires. The MAX TNT sends out IGMP queries every 60 seconds, so the expiration time is usually renewed. If the expiration time is reached, the MAX TNT removes the entry from the table. If the field contains periods, this membership never expires.
`Counts`	Number of packets forwarded to the client, number of packets dropped due to lack of resources, and the state of the membership. The state is displayed for debugging purposes.

Displaying IGMP client information

To display a list of multicast clients, enter the IGMP command with the `client` option:

```
admin> igmp client
IGMP Clients
Client      Version  RecvCount  CLU      ALU
  0(Mbone)      1       0          0       0
  2             1      39         68       67
  1             1   33310         65       65
```

The output contains the following fields:

Field	Description
`Client`	ID of the interface on which the client resides. The value 0 (zero) represents the Ethernet. Other numbers are WAN interfaces, numbered according to when they became active. `Mbone` is the interface on which the multicast router resides.
`Version`	Version of IGMP being used.
`RecvCount`	Number of IGMP messages received on the client's interface.

Field	Description
CLU ALU	Current Line Utilization and Average Line Utilization, respectively. Both indicate the percentage of bandwidth utilized across this interface. If bandwidth utilization is high, some IGMP packet types are not forwarded.

Diagnostic tools for OSPF routers

The OSPF diagnostic-level commands enable the administrator to display information related to OSPF routing, including the link state advertisements (LSAs), border routers' routing table, and the OSPF areas, interfaces, statistics, and routing table. To display the usage statement, enter the OSPF command with the ? option:

```
admin> ospf

ospf ?                              OSPF help information
ospf size                           OSPF size
ospf areas                          OSPF areas
ospf stats                          OSPF statistics
ospf intf [ip-address]              OSPF summary/detail interface information
ospf lsa area ls-type ls-id ls-orig OSPF detail link-state advertisement
ospf lsdb [area]                    OSPF link-state DB summary for an area
ospf nbrs [neighbor-id]             OSPF summary/detail neighbor information
ospf routers                        OSPF routers
ospf ext                            OSPF external AS advertisements
ospf rtab                           OSPF routing table
ospf database                       OSPF entire database summary
ospf internal                       OSPF internal routes
```

Displaying general information about OSPF routing

To display general information about OSPF, enter the OSPF command with the `stat` option. For example:

```
admin> ospf stats

OSPF version:           2
               OSPF Router ID:         10.103.0.254
               AS boundary capability: Yes
  Attached areas:                    1  Estimated # ext.(5) routes:   65536
  OSPF packets rcvd:             71788  OSPF packets rcvd w/ errs:       19
  Transit nodes allocated:         812  Transit nodes freed:            788
  LS adv. allocated:              2870  LS adv. freed:                 2827
  Queue headers alloc:              64  Queue headers avail:             64
  # Dijkstra runs:                  10  Incremental summ. updates:        0
  Incremental VL updates:            0  Buffer alloc failures:           0
  Multicast pkts sent:           27343  Unicast pkts sent:              1154
  LS adv. aged out:                  0  LS adv. flushed:                507
  Incremental ext.(5) updates:    1014  Incremental ext.(7) updates:      0
  External (Type 5) LSA database -
  Current state:                 Normal
  Number of LSAs:                   43
  Number of overflows:               0
```

The following table describes the output:

Field	Specifies
`OSPF version`	Version of the OSPF protocols running.
`OSPF Router ID`	IP address assigned to the MAX TNT, which is typically the address specified for the Ethernet interface.
`AS boundary capability`	Yes if the MAX TNT functions as an ASBR or No if it does not function as an ASBR.
`Attached areas`	Number of areas to which this MAX TNT attaches.
`Estimated # ext.(5) routes`	Number of ASE-5 routes that the MAX TNT can maintain before it goes into an overload state.
`OSPF packets rcvd`	Total number of OSPF packets received by the MAX TNT.
`OSPF packets rcvd w/ errs`	Total number of OSPF errored packets received by the MAX TNT.
`Transit nodes allocated`	Allocated transit nodes generated only by Router LSAs (Type 1) and Network LSAs (Type 2).
`Transit nodes freed`	Freed transit nodes generated only by Router LSAs (Type 1) and Network LSAs (Type 2).
`LS adv. allocated`	Number of LSAs allocated.
`LS adv. freed`	Number of LSAs freed.
`Queue headers alloc`	Number of queue headers allocated. LSAs can reside in multiple queues. Queue headers are the elements of the queues that contain the pointer to the LSA.
`Queue headers avail`	Available memory for queue headers. To prevent memory fragmentation, the MAX TNT allocates memory in blocks. The MAX TNT allocates queue headers from the memory blocks. When the MAX TNT frees all queue headers from a specific memory block, the MAX TNT returns the block to the pool of available memory blocks.
`# Dijkstra runs`	Number of times that the MAX TNT has run the Dijkstra algorithm (short path computation).
`Incremental summ. updates`	Number of summary updates that the MAX TNT runs when small changes cause generation of Summary LSAs (Type 3) and Summary Router LSAs (Type 4).
`Incremental VL updates`	Number of incremental virtual link updates that the MAX TNT performs.
`Buffer alloc failures`	Number of buffer allocation problems that the MAX TNT has detected and from which it has recovered.
`Multicast pkts sent`	Number of multicast packets sent by OSPF.
`Unicast pkts sent`	Number of unicast packets sent by OSPF.
`LS adv. aged out`	Number of LSAs that the MAX TNT has aged and removed from its tables.

Field	Specifies
`LS adv. flushed`	Number of LSAs that the MAX TNT has flushed.
`Incremental ext.(5) updates`	Number of incremental ASE-5 updates.
`Incremental ext.(7) updates`	Number of incremental ASE-7 updates.
`Current state`	State of the External (Type-5) LSA database: Normal or Overload.
`Number of LSAs`	Number of LSAs in the External (Type-5) LSA database.
`Number of overflows`	Number of ASE-5s that exceeded the limit of the database.

Displaying the OSPF database

To display the entire OSPF database, enter the OSPF command with the `database` option. For example:

```
admin> ospf database

                    Router Link States (Area: 0.0.0.0)
Type LS ID              LS originator     Seqno     Age   Xsum
RTR  10.101.0.1         10.101.0.1      0x800002a1  746  0x8bd8
RTR  10.101.0.2         10.101.0.2      0x800002d6  539  0x0ea1
RTR  10.102.0.1         10.102.0.1      0x800002a3 2592  0x9bc1
RTR  10.103.0.204       10.103.0.204    0x800001ba 1173  0x725f
RTR  10.103.0.254       10.103.0.254    0x80000301  534  0x7066
RTR  10.104.0.1         10.104.0.1      0x800002ad  777  0xb98e
RTR  10.104.0.2         10.104.0.2      0x80000193 1258  0x265a
RTR  10.105.0.2         10.105.0.2      0x80000299  865  0x4295
RTR  10.105.0.3         10.105.0.3      0x800002e5 1057  0x4449
RTR  10.105.0.4         10.105.0.4      0x80000310 1585  0x5775
RTR  10.105.0.61        10.105.0.61     0x800002ae 1204  0xcf2e
RTR  10.105.0.200       10.105.0.200    0x80000263  213  0x4b25
RTR  10.123.0.8         10.123.0.8      0x80000401 1071  0xecf2
RTR  10.123.0.254       10.123.0.254    0x80000401 1175  0xad39
RTR  12.151.0.2         12.151.0.2      0x800006ee  825  0x0531
RTR  192.1.1.1          192.1.1.1       0x8000039b   18  0xb04b
RTR  210.210.210.1      210.210.210.1   0x800001aa  201  0x5338
               # advertisements:       17
               Checksum total:         0x7946c

                    Network Link States (Area: 0.0.0.0)
Type LS ID              LS originator     Seqno     Age   Xsum
NET  10.101.0.1         10.101.0.1      0x80000236  746  0x1d45
NET  10.102.0.1         10.102.0.1      0x80000235 2592  0x1f40
NET  10.104.0.2         10.104.0.2      0x80000179  830  0x67a8
NET  10.105.0.8         10.123.0.8      0x80000304 1071  0x0ccd
NET  10.123.0.6         12.151.0.2      0x8000023d  825  0x59ed
NET  100.103.100.204    10.103.0.204    0x80000029  252  0x8b34
               # advertisements:       6
```

```
                Checksum total:         0x1961b

                        External ASE5 Link States
Type LS ID              LS originator     Seqno     Age   Xsum
ASE5 10.103.1.0         10.103.0.204    0x8000004f 1726  0xd23f
ASE5 10.103.2.0         10.103.0.204    0x8000004f 1716  0xc749
ASE5 10.103.3.0         10.103.0.204    0x8000004f 1704  0xbc53
ASE5 10.103.4.0         10.103.0.204    0x8000004f 1692  0xb15d
ASE5 10.103.6.0         10.103.0.204    0x8000004f 1672  0x9b71
ASE5 10.103.7.0         10.103.0.204    0x8000004f 1666  0x907b
ASE5 10.103.8.0         10.103.0.204    0x8000004f 1641  0x8585
ASE5 10.107.0.0         10.103.0.254    0x80000104  250  0x1413
ASE5 10.113.0.0         10.103.0.254    0x80000121  250  0x0e76
ASE5 10.200.0.2         10.103.0.254    0x80000001  231  0xa823
ASE5 10.222.0.2         10.103.0.254    0x80000001  202  0x9f16
ASE5 11.0.0.0           10.103.0.254    0x80000027  250  0x49a6
ASE5 11.103.0.0         10.103.0.254    0x80000121  250  0xfc10
ASE5 14.240.0.0         10.103.0.204    0x800001a4  199  0x0926
ASE5 50.151.0.2         10.103.0.254    0x80000121  250  0xa90a
ASE5 101.103.0.0        10.103.0.254    0x80000121  250  0x664c
..
..
                # advertisements:       44
                Checksum total:         0x191d3a
```

The following table describes the output:

Field	Specifies
`Type`	Type of link as defined in RFC 1583: • Type 1 (RTR) are router-LSAs that describe the collected states of the router's interfaces. • Type 2 (NET) are network-LSAs that describe the set of routers attached to the network. • Types 3 and 4 (SUM) describe routes to networks in remote areas or AS boundary routers. • Type 5 (ASE) are AS-external-LSAs that describe routes to destinations external to the Autonomous System. A default route for the Autonomous System can also be described by an AS-external-LSA. The `ext` option only displays ASE5 LSAs. • Type 7 are ASE-7 link advertisements that are only flooded within an NSSA.
`LS ID`	Target address of the route.
`LS originator`	Address of the advertising router.
`Seqno`	Hexadecimal number that begins with 80000000 and increments by one for each LSA received.
`Age`	Age of the route in seconds.
`Xsum`	Checksum of the LSA.

Field	Specifies
`# advertisements`	Total number of entries in the database.
`Checksum total`	Checksum of the database.

Displaying OSPF external AS advertisements

To display only OSPF External AS advertisements, include the `ext` option with the OSPF command. For example:

```
admin> ospf ext
Type LS ID              LS originator     Seqno     Age   Xsum
ASE5 10.103.1.0         10.103.0.204    0x8000004f 1702  0xd23f
ASE5 10.103.2.0         10.103.0.204    0x8000004f 1692  0xc749
ASE5 10.103.3.0         10.103.0.204    0x8000004f 1680  0xbc53
ASE5 10.103.4.0         10.103.0.204    0x8000004f 1668  0xb15d
ASE5 10.103.6.0         10.103.0.204    0x8000004f 1648  0x9b71
ASE5 10.103.7.0         10.103.0.204    0x8000004f 1642  0x907b
ASE5 10.103.8.0         10.103.0.204    0x8000004f 1617  0x8585
..
..
ASE5 214.240.0.127      10.103.0.204    0x800001a4  175  0xdb0b
ASE5 223.57.40.0        10.103.0.254    0x80000121  226  0x7540
ASE5 223.57.40.244      10.103.0.254    0x80000121  226  0xe3dc
               # advertisements:       46
               Checksum total:         0x1a1d9e
```

The output of this command is the same as for the OSPF database command, with the exception of the Type. The OSPF Ext command only shows ASE5 type LSAs.

Displaying OSPF internal AS advertisements

To display OSPF internal AS advertisements, include the `internal` option with the OSPF command. For example:

```
admin> ospf internal
                Area: 0.0.0.1
Destination     Mask            Cost
33.240.0.0      255.255.255.224  1
103.240.0.0     255.255.255.192  1
113.240.0.0     255.255.255.128  1
183.240.0.0     255.255.255.128  1
193.240.0.0     255.255.255.128  1
203.240.0.0     255.255.255.128  1
```

The following table describes the output:

Field	Specifies
`Area`	Area in which the router resides.

Field	Specifies
`Destination`	The route's target address. To send a packet to this address, the MAX TNT uses this route. If the target address appears more than once in the routing table, the MAX TNT uses the most specific route (having the largest subnet mask) that matches that address.
`Mask`	Subnet mask of the route.
`Cost`	Cost of the router.

Displaying the OSPF link-state database

To display the link-state database for the first configured area (or for the only defined area), include the `lsdb` option with the OSPF command. The MAX TNT does not currently operate as an ABR, so each MAX TNT OSPF interface belongs to the same area. (That area number does not have to be the default backbone area 0.0.0.0.)

For example:

```
admin> ospf lsdb

                    Area: 0.0.0.0
Type LS ID              LS originator     Seqno     Age   Xsum
RTR  10.101.0.1         10.101.0.1      0x8000029f  720  0x8fd6
RTR  10.101.0.2         10.101.0.2      0x800002d1  126  0x189c
RTR  10.102.0.1         10.102.0.1      0x800002a2  767  0x9dc0
RTR  10.102.0.2         10.102.0.2      0x800002cc  124  0x862c
RTR  10.103.0.204       10.103.0.204    0x800001b8 1147  0x765d
RTR  10.103.0.254       10.103.0.254    0x800002fb  167  0x8cc9
RTR  10.104.0.1         10.104.0.1      0x800002ab  751  0xbd8c
RTR  10.104.0.2         10.104.0.2      0x80000191 1232  0x2a58
RTR  10.105.0.2         10.105.0.2      0x80000297  843  0x4693
RTR  10.105.0.3         10.105.0.3      0x800002e3 1032  0x4847
RTR  10.105.0.4         10.105.0.4      0x8000030e 1560  0x5b73
RTR  10.105.0.61        10.105.0.61     0x800002ac 1178  0xd32c
RTR  10.105.0.200       10.105.0.200    0x80000261  194  0x4f23
RTR  10.123.0.8         10.123.0.8      0x800003ff 1045  0xf1ef
RTR  10.123.0.254       10.123.0.254    0x800003ff 1149  0xb236
RTR  12.151.0.2         12.151.0.2      0x800006ec  799  0x092f
RTR  192.1.1.1          192.1.1.1       0x80000398 1791  0xb648
RTR  210.210.210.1      210.210.210.1   0x800001a8  175  0x5736
NET  10.101.0.1         10.101.0.1      0x80000234  720  0x2143
NET  10.102.0.1         10.102.0.1      0x80000234  767  0x213f
NET  10.104.0.2         10.104.0.2      0x80000177  804  0x6ba6
NET  10.105.0.8         10.123.0.8      0x80000302 1045  0x10cb
NET  10.123.0.6         12.151.0.2      0x8000023b  799  0x5deb
NET  100.103.100.204    10.103.0.204    0x80000027  226  0x8f32
                # advertisements:       24
                Checksum total:         0xa2ae6
```

The fields in the output contain the following information:

Field	Specifies
`Area`	Area ID.
`Type`	Indicates the type of link as defined in RFC 1583: • Type 1 (RTR) are router-LSAs that describe the collected states of the router's interfaces. • Type 2 (NET) are network-LSAs that describe the set of routers attached to the network. • Types 3 and 4 (SUM) describe routes to networks in remote areas or AS boundary routers. • Type 7 are ASE-7 link advertisements that are only flooded within an NSSA.
`LS ID`	Specifies the target address of the route.
`LS originator`	Specifies the address of the advertising router.
`Seqno`	Indicates a hexadecimal number that begins with 80000000 and increments by one for each LSA received.
`Age`	Specifies the age of the route in seconds.
`Xsum`	Indicates the checksum of the LSA.
`advertisements`	Specifies the total number of entries in the link-state database.
`Checksum total`	Indicates the checksum of the link-state database.

You can expand each entry in the link-state database to view additional information about a particular LSA, as explained in the next section.

Displaying OSPF link-state advertisements

To view detailed information about a link-state advertisement, use the following format for the OSPF command:

`ospf lsa` *`area ls-type ls-id ls-orig`*

The command requires that you include the first four fields of the LSA as listed in the database. You can select the first four fields and paste them in after typing the command. For example, to show an expanded view of the last entry in the link-state database shown in the previous section:

```
admin> ospf lsa 0.0.0.0 ase 10.5.2.160 10.5.2.162

LSA  type: ASE ls id: 10.5.2.160 adv rtr: 110.5.2.162 age: 568
          seq #: 80000037 cksum: 0xfffa
          Net mask: 255.255.255.255 Tos 0 metric: 10 E type: 1
          Forwarding Address: 0.0.0.0 Tag: c0000000
```

The output differs depending on the type of link. The following is an example of a router LSA:

```
admin> ospf lsa 0.0.0.0 rtr 192.1.1.1 192.1.1.1
```

```
        LS age:      66
        LS options:  (0x2) E
        LS type:     1
        LS ID (destination): 192.1.1.1
        LS originator:       192.1.1.1
        LS sequence no:      0x80000399
        LS checksum:         0xb449
        LS length:           48
        Router type:     (0x2) ASBR
        # router ifcs:   2
                Link ID:          10.105.0.8
                Link Data:        10.105.0.7
                Interface type:   (2) TrnsNetwork
                        No. of metrics: 0
                        TOS 0 metric:   10 (0)
                Link ID:          10.123.0.6
                Link Data:        10.123.0.7
                Interface type:   (2) TrnsNetwork
                        No. of metrics: 0
                        TOS 0 metric:   10 (0)
```

The next example is for a network LSA:

```
admin> ospf lsa 0.0.0.0 net 100.103.100.204 10.103.0.204
        LS age:      814
        LS options:  (0x2) E
        LS type:     2
        LS ID (destination): 100.103.100.204
        LS originator:       10.103.0.204
        LS sequence no:      0x80000027
        LS checksum:         0x8f32
        LS length:           36
        Network mask:        255.255.0.0
                Attached Router: 10.103.0.204     (1)
                Attached Router: 10.103.0.254     (1)
                Attached Router: 10.123.0.254     (1)
```

For information about the fields in the output of these commands, see the *MAX TNT Reference Guide* or RFC 1583.

Displaying the OSPF routing table

To display the OSPF routing table, include the `rtab` option with the OSPF command. For example:

```
admin> ospf rtab
DTyp RType Destination        Area           Cost  Flags  Next hop(s)     IfNum
RTE  FIX   50.151.0.2/32      -              1     0x81   0.0.0.6         6
RTE  FIX   130.57.40.243/32   -              10    0x1    0.0.0.6         6
RTE  FIX   130.57.0.0/16      -              10    0x2    0.0.0.6         6
RTE  FIX   140.57.40.244/32   -              10    0x1    0.0.0.6         6
RTE  FIX   140.57.0.0/16      -              10    0x2    0.0.0.6         6
RTE  FIX   150.57.40.245/32   -              10    0x1    0.0.0.6         6
RTE  FIX   150.57.0.0/16      -              10    0x2    0.0.0.6         6
```

```
RTE  FIX   160.57.40.246/32   -              10    0x1    0.0.0.6         6
RTE  FIX   160.57.0.0/16      -              10    0x2    0.0.0.6         6
..
..
..
```

The fields in the output contain the following information:

Field	Specifies
DType	Internal route type. DType displays one of the following values: RTE (generic route), ASBR (AS border route), or BR (area border route).
RType	Internal router type. RType displays one of the following values: FIX (static route), NONE, DEL (deleted or bogus state), OSPF (OSPF-computed), OSE1 (type 1 external), or OSE2 (type 2 external).
Destination	Destination address and subnet mask of the route.
Area	Area ID of the route.
Cost	Cost of the route.
Flags	Hexadecimal number representing an internal flag.
Next hop(s)	Next hop in the route to the destination.
#	Number of the interface used to reach the destination.

The fields in the output contain the following information:

Field	Specifies
LSA type	Type of Link-State Advertisement.
ls id	Target address of the router.
adv rtr	Address of the advertising router.
age	Age of the route in seconds.
seq #	Number that begins with 80000000 and increments by one for each LSA received.
cksum	Checksum for the LSA.
Net mask	Subnet mask of the LSA.
Tos	Type of Service for the LSA.
metric	Cost of the link, not of a route. The cost of a route is the sum of all intervening links, including the cost of the connected route.
E type	External type of the LSA indicating either 1 (Type 1) or 2 (Type 2)
Forwarding Address	Forwarding Address of the LSA (described in RFC 1583).
Tag	Tag of the LSA (described in RFC 1583).

Displaying information about OSPF areas

To display information about OSPF areas, include the `areas` option with the OSPF command. For example:

```
admin> ospf areas
Area ID  Authentication  Area Type #ifcs  #nets  #rtrs  #brdrs  #intnr
0.0.0.0  Simple-passwd   Normal      1      0      2       0       3
```

The fields in the output contain the following information:

Field	Specifies
`Area ID`	Area number in dotted-decimal format.
`Authentication`	Type of authentication: Simple-passwd, MD5, or Null.
`Area Type`	Type of OSPF area: Normal, Stub, or NSSA.
`#ifcs`	Number of MAX TNT interfaces specified in the area.
`#nets`	Number of reachable networks in the area.
`#rtrs`	Number of reachable routers in the area.
`#brdrs`	Number of reachable area border routers in the area.
`#intnr`	Number of reachable internal routers in the area.

Displaying information about OSPF routers

To display OSPF routers, include the `routers` option with the OSPF command. For example:

```
admin> ospf routers
DType RType Destination       Area           Cost       Next hop(s)     IfNum
ASBR  OSPF  10.101.0.1       0.0.0.0         11        10.101.0.2      20
ASBR  OSPF  10.101.0.2       0.0.0.0         10        10.101.0.2      20
ASBR  OSPF  10.103.0.204     0.0.0.0         1         100.103.100.204 24
ASBR  OSPF  10.104.0.1       0.0.0.0         12        10.105.0.4      21
                                                       10.105.0.61     21
ASBR  OSPF  10.104.0.2       0.0.0.0         11        10.105.0.4      21
                                                       10.105.0.61     21
  BR  OSPF  10.105.0.2       0.0.0.0         1         10.105.0.2      21
ASBR  OSPF  10.105.0.2       0.0.0.0         1         10.105.0.2      21
ASBR  OSPF  10.105.0.3       0.0.0.0         1         10.105.0.3      21
ASBR  OSPF  10.105.0.4       0.0.0.0         1         10.105.0.4      21
ASBR  OSPF  10.105.0.61      0.0.0.0         1         10.105.0.61     21
ASBR  OSPF  10.105.0.200     0.0.0.0         1         10.105.0.200    21
ASBR  OSPF  10.123.0.8       0.0.0.0         1         10.105.0.8      21
ASBR  OSPF  10.123.0.254     0.0.0.0         1         100.103.100.123 24
  BR  OSPF  12.151.0.2       0.0.0.0         1         10.105.0.6      21
ASBR  OSPF  192.1.1.1        0.0.0.0         1         10.105.0.7      21
```

The fields in the output contain the following information:

Field	Specifies
DType	Internal route type. DType displays one of the following values: RTE (generic route), ASBR (AS border route), or BR (area border route).
RType	Internal router type.
Destination	Router's IP address.
Area	Area in which the router resides.
Cost	Cost of the router.
Next hop(s)	Next hop in the route to the destination.
IfNum	Number of the interface used to reach the destination.

Displaying OSPF interfaces

To display either summarized information about all OSPF interfaces or specific information about a single interface, include the `intf` option with the OSPF command.

Displaying summarized information

To display summarized information on OSPF interfaces, enter the following command:

```
admin> ospf intf
```

```
Ifc Address     Phys     Assoc. Area     Type   State    #nbrs #adjs DInt
10.103.0.254    ie0      0.0.0.0         Brdcst DR        0     0    40
10.105.0.254    ie1-7-1  0.0.0.0         Brdcst Other     9     1    40
100.103.100.254 ie1-7-4  0.0.0.0         Brdcst Other     2     2    40
50.151.0.2      tnt1     0.0.0.0         P-P    P-P       0     0    120
10.103.0.254    m2       0.0.0.0         P-P    P-P       1     1    120
10.103.0.254    m1       0.0.0.0         P-P    P-P       1     1    120
```

The fields in the output contain the following information:

Field	Specifies
Ifc Address	Address assigned to the MAX TNT's Ethernet interface. To identify WAN links, use the Type and Cost fields.
Phys	Name of the interface or the Connection profile for WAN links.
Assoc. Area	Area in which the interface resides.
Type	Point-to-Point (P-P) or Broadcast (Brdcst). WAN links are P-P links.
State	State of the link according to RFC 1583. There are many possible states, and not all states apply to all interfaces.
#nbrs	Number of neighbors of the interface.
#adjs	Number of adjacencies on the interface.

Field	Specifies
DInt	Number of seconds that the MAX TNT waits for a router update before removing the router's entry from its table. The interval is called the Dead Interval.

Displaying specific information about a specific interface

To display detailed information for a specific interface, enter the following command:

```
admin> ospf intf interface-address
```

For example:

```
admin> ospf intf 194.194.194.2
             Interface address:      194.194.194.2
             Attached area:          0.0.0.0
             Physical interface:     phani (wan1)
             Interface mask:         255.255.255.255
             Interface type:         P-P
             State:                  (0x8) P-P
             Designated Router:      0.0.0.0
             Backup DR:              0.0.0.0
             Remote Address:         194.194.194.3
DR Priority:       5  Hello interval:  30   Rxmt interval:  5
Dead interval:   120  TX delay:         1   Poll interval:  0
Max pkt size:   1500  TOS 0 cost:      10
# Neighbors:       1  # Adjacencies:    1   # Full adjs.:   1
# Mcast floods: 1856  # Mcast acks:  1855
```

The fields in the output contain the following information:

Field	Specifies
Interface Address	IP address specified for the MAX TNT unit's Ethernet interface.
Attached Area	Area in which the interface resides.
Physical interface	Name of the interface or the Connection profile for WAN links.
Interface type	Point-to-Point (P-P) or Broadcast (Bcast). WAN links are P-P links.
State	State of the link according to RFC 1583. There are many possible states, and not all states apply to all interfaces.
Designated Router	IP address of the designated router for the interface.
Backup DR	IP address of the backup designated router for the interface.
Remote Address	IP address of the remote end of a Point to Point (WAN) link.
DR Priority	Priority of the designated router.
Hello interval	Interval in seconds that the MAX TNT sends Hello packets (as defined in RFC 1583).

Field	Specifies
`Rxmt interval`	Retransmission interval (as described in RFC 1583).
`Dead interval`	Number of seconds that the MAX TNT waits for a router update before removing the router's entry from its table.
`TX delay`	Interface transmission delay.
`Poll interval`	Poll interval of nonbroadcast multiaccess networks.
`Max pkt size`	Maximum size of a packet that the MAX TNT can send to the interface.
`TOS 0 Count`	Type of Service normal (*0*) cost.
`# neighbors`	Number of neighbors.
`# adjacencies`	Number of adjacencies.
`# Full adjs.`	Number of fully-formed adjacencies.
`# Mcast floods`	Number of multicast floods on the interface.
`# Mcast acks`	Number of multicast acknowledgments on the interface.

Displaying OSPF neighbors

To display information about OSPF neighbors to the MAX TNT, include the `nbrs` option with the OSPF command. For example:

```
admin> ospf nbrs

Neighbor ID     Neighbor addr   State      LSrxl DBsum LSreq Prio Ifc
10.105.0.4      10.105.0.4      2Way/-       0     0     0     5  ie1-7-1
10.105.0.2      10.105.0.2      2Way/-       0     0     0     5  ie1-7-1
12.151.0.2      10.105.0.6      2Way/-       0     0     0     1  ie1-7-1
10.105.0.3      10.105.0.3      2Way/-       0     0     0     5  ie1-7-1
10.105.0.61     10.105.0.61     2Way/-       0     0     0     5  ie1-7-1
210.210.210.1   10.105.0.49     Exstar/BDR   0     0     0     5  ie1-7-1
192.1.1.1       10.105.0.7      2Way/-       0     0     0     5  ie1-7-1
10.123.0.8      10.105.0.8      Full/DR      0     0     0     5  ie1-7-1
10.105.0.200    10.105.0.200    2Way/-       0     0     0     5  ie1-7-1
10.103.0.204    100.103.100.204 Full/DR      0     0     0     5  ie1-7-4
10.123.0.254    100.103.100.123 Full/BDR     0     0     0     5  ie1-7-4
10.102.0.2      10.102.0.2      Init/-       0     0     0     5  m1
10.101.0.2      10.101.0.2      Full/-       0     0     0     5  m1
```

The fields in the output contain the following information:

Field	Specifies
`Neighbor ID`	Address assigned to the interface. In the MAX TNT, the IP address is always the address assigned to the Ethernet interface.
`Neighbor addr`	IP address of the router used to reach a neighbor (often the same address as the neighbor itself).

Field	Specifies
`State`	State of the link-state database exchange. Full indicates that the databases are fully aligned between the MAX TNT and its neighbor. For a description of possible states, see RFC 1583.
`LSrxl`	Number of LSAs in the retransmission list.
`DBsum`	Number of LSAs in the database summary list.
`LSreq`	Number of LSAs in the request list.
`Prio`	Designated router election priority assigned to the MAX TNT.
`Ifc`	Interface name for the Ethernet or Connection profile name for the WAN.

To display information about a particular OSPF neighbor, append the Neighbor ID to the `nbrs` option. For example:

```
admin> ospf nbrs 10.105.0.4

OSPF Router ID:         10.105.0.4
               Neighbor IP address:    10.105.0.4
               Neighbor State:         (0x8) 2Way
               Physical interface:     ie1-7-1 (ie1-7-1)
               DR choice:              10.105.0.8
               Backup choice:          10.105.0.49
               DR Priority:            5
  DB summ qlen:      0  LS rxmt qlen:      0  LS req qlen:       0
  Last hello:        6
  # LS rxmits:       0  # Direct acks:     0  # Dup LS rcvd:     0
  # Old LS rcvd:     0  # Dup acks rcv:    0  # Nbr losses:      0
  # Adj. resets:     0
```

Diagnostic tools for IPX routers

The MAX TNT provides two diagnostic commands for monitoring IPX networks, Show Netware Servers and Show Netware Networks.

To display the IPX service table, first enter the Terminal-Server command to access the MAX TNT terminal server interface, then enter the Show command with the `netware servers` option. For example:

```
admin> terminal-server

** Ascend TNT Terminal Server **

ascend% show netware servers

IPX address                    type             server name
ee000001:000000000001:0040     0451             server-1
```

The output contains these fields:

- `IPX address`: The IPX address of the server. The address uses this format:
  network number:*node number*:*socket number*

- `type`: The type of service available (in hexadecimal format). For example, 0451 designates a file server.
- `server name`: The first 35 characters of the server name.

To display the IPX routing table, enter the Show command with the `netware networks` option. For example:

```
ascend% show netware networks

network     next router        hops        ticks   origin
CFFF0001    00000000000        0         1       Ethernet        S
```

The output contains these fields:

Fields	Descriptions
`network`	The IPX network number.
`next router`	The address of the next router, or 0 (zero) for a direct or WAN connection.
`hops`	The hop count from the shelf controller to the network.
`ticks`	The tick count to the network.
`origin`	The name of the profile used to reach the network. If the origin is a network connected to a MAX TNT Ethernet interface, the Origin field displays `Ethernet`.

Note: An `S` or an `H` flag can appear next to the origin. `S` indicates a static route. `H` indicates a hidden static route. Hidden static routes occur when the router learns of a better route.

Displaying Ethernet packet contents

The Ether-Display command displays the hexadecimal contents of Ethernet packets being received and transmitted on the specified Ethernet port. You must specify how many octets of each packet you want to display.

The Ether-Display command requires that you enable debug output as follows:

```
admin> debug on
Diagnostic output enabled
```

The following example displays 12 octets of each packet on a ports:

```
admin> ether-display 0 12
ETHER XMIT: 12 of 60 octets
107E1350: 00 c0 80 89 03 d7 00 c0  7b 6b 9f d6                ........ {k..
ETHER XMIT: 12 of 64 octets
107E1350: 00 c0 80 89 03 d7 00 c0  7b 6b 9f d6                ........ {k..
ETHER RECV: 12 of 60 octets
107B8FD4: 00 c0 7b 6b 9f d6 00 c0  80 89 03 d7                ..{k.... ....
ETHER XMIT: 12 of 407 octets
107E1350: 00 c0 80 89 03 d7 00 c0  7b 6b 9f d6                ........ {k..
ETHER XMIT: 12 of 161 octets
107E1350: 00 c0 80 89 03 d7 00 c0  7b 6b 9f d6                ........ {k..
```

```
ETHER RECV: 12 of 60 octets
..
..
..
```

To stop displaying the Ethernet statistics, specify 0 (zero) octets:

```
admin> ether-display 0 0
```

Alternatively, you can stop the display by disabling debug output:

```
admin> debug off
Diagnostic output disabled
```

For complete information about the Ether-Display command, see the *MAX TNT Reference Guide*.

Using the MAX TNT Debug Commands 5

This chapter covers the following topics:

Note: Every attempt has been made to confirm that this chapter correctly describes the functionality and output of the MAX TNT debug commands. However, while debug mode can be a very valuable troubleshooting tool for anyone, its primary focus is on the requirements of Ascend's development engineers. For this reason, Ascend does not guarantee the completeness of the list of commands published for a given release nor the exhaustive cataloging of their functionality.

Caution: Under most circumstances, debug commands are not required for correct operation of the MAX TNT. And in some circumstances they might produce undesirable results. Please use the following information with caution. Contact Ascend Technical Support with any questions or concerns.

Enabling debug permissions

Before you can access the debug commands, you must log into the MAX TNT with a User profile that specifies debug privileges.

To enable debugging privileges:

1 Open a user profile:

```
admin> open user admin
```

2 Enable debug permissions:

```
admin> set allow-debug=yes
```

This is a hidden parameter. It does not appear in the interface.

3 Write the profile to save the changes:

```
admin> write
```

Note that when you are logged into the MAX TNT with debug privileges, the interface might display normally unavailable parameters and commands, some of which are not configurable in certain situations. For this reason, you should create a special profile for debugging purposes, and only use that profile when you are debugging the MAX TNT.

Enabling debug output

To enable debug output for all commands on the system or on a card, use the Debug command as in the following examples.

To enable debug:

```
hdlc-2/1> debug on
Diagnostic output enabled
```

To disable debug:

```
hdlc-2/1> debug off
Diagnostic output disabled
```

When you enable debug output, the MAX TNT displays the debug messages on the terminal screen.

Debug levels

Debug levels vary depending on the command. But generally, the lower you set the debug level, the fewer messages the MAX TNT displays. Setting the debug level to 0 (zero) disables the debug output for the command.

Set the debug level with the `-t` option, as in the following examples:

```
admin> ifmgr -t 0
ifmgr debug level is now 0 (disabled)


admin> ifmgr -t 4
ifmgr debug level is now 4 (enabled)
```

Getting online help for debug commands

To see a list of all commands, including the debug commands, enter `?` at the command prompt, as in the following example:

```
admin> ?
 ?                              ( user )
 @fatalTest                     ( debug )
 acctevnt                       ( debug )
 addrpool                       ( debug )
 ARA                            ( debug )
 aracbmgr                       ( debug )
 arptable                       ( system )
 atmpdebug                      ( debug )
```

```
 auth                           ( user )
 briChannels                    ( system )
 brouterDebug                   ( debug )
 brouterLoad                    ( debug )
 brouterMessage                 ( debug )
 brouterSave                    ( debug )
 brouterstats                   ( debug )
 cadslLines                     ( system )
 callback                       ( debug )
 callblocks                     ( debug )
 callroute                      ( diagnostic )
 cbacctevnt                     ( debug )
 cbcardif                       ( debug )
 cbcifping                      ( debug )
[More? <ret>=next entry, <sp>=next page, <^C>=abort]
```

To get basic help for a debug command, enter the Help command, followed by the name of the debug command, as in the following example:

```
admin> help ifmgr
ifmgr usage: ifmgr -option
             -d  (d)isplay interface table entries.
             -d  <ifNum> (d)etails of given i/f table entry.
             -t  (t)oggle debug display.
       ifmgr [up|down] [ifNum|ifName]
```

Using combinations of commands

Since most debug commands are designed to give a developer information about specific portions of MAX TNT functionality, you might find it helpful to use commands in combination to troubleshoot different problems.

For example, if you see problems with the initial connection of remote users, you might want to use a combination of `Networki`, `Routmgr` and `Wantoggle` to obtain a complete view of three functions involved in establishing a call.

When troubleshooting modem-related issues, you might want to use `Modemdrvstate`, `Modemdiag` and `Mdialout` (if modem outdial is supported on your MAX TNT) to get all modem-related information for your calls.

Using several commands simultaneously not only gives you a clearer picture of a given situation, it also shows you a chronological timeline of the events that are happening.

Using the debug commands

Debug commands allow you to monitor and diagnose different areas of the MAX TNT functionality. This section lists some of the more common debug commands and the areas of the MAX TNT they apply to.

Frame Relay

The following commands display information about Frame Relay interfaces.

- FRDLstate
- FRdump
- FRinARP
- FRLinkState
- FRLMI
- FRMgrDump
- FRPriorityErrors
- FRScert
- FRstate

Calls

The following commands display information about how the MAX TNT handles calls.

- Callback
- Permconn-list
- Tntcall
- Routmgr

Authentication

The following commands display information about how the MAX TNT authenticates calls.

- Authendebug
- Lanval
- Radacct
- Raddbgdump
- Radif
- Radservdump
- Radsessdump
- Radstats

Multishelf

The following commands display information about the MAX TNT multishelf system.

- Cubit
- Msstat
- Pbecho
- Sar
- Tdm
- Tdmtest

Host-side devices

The following commands display information about the MAX TNT host devices.

- ModemDrvDump
- ModemDrvState
- Modemd1stats, Modemd2stats, Modemd3stats
- Ether-Stats
- Ifmgr

Network-side devices

The following commands display information about the MAX TNT network devices.

- NetIF
- Networki
- Pridisplay
- WANdisplay
- WanEventsStats
- WANopening
- Wantoggle

Protocols

The following commands display information about MAX TNT protocols.

- Addrpool
- Brouterdebug
- Brouterload
- Ctcheck
- Ctdebug
- Ipxripdebug
- Lcstate
- Leakpool
- Ospfavltree
- Ospfdebug
- Sntp

- Tcpflushtimer

Tunneling

The following commands display information about MAX TNT tunneling.

- ATMP
- Dtunnel
- Tunneldebug
- Tunnelslot

System and devices

The following commands display information about the MAX TNT system and devices.

- Pools
- Portinfo
- Reset
- Revision
- Stacklimit
- Stackusage
- Tsshow
- Update
- Watchdogtoggle

Terminal server

The following commands display information about the MAX TNT terminal server.

- Telnetdebug
- Tsbadterminfo

Special administrative commands

The following command should only be used when requested to by Ascend technical support.

- Coredump

Alphabetical list of debug commands

This section describes the MAX TNT debug commands in alphabetic order. The information is organized for quick reference, and does not include tutorials.

Acct-Failsafe

Description: The Acct-Failsafe debug command is available on the master shelf or the slot host cards for verifying correct accounting proxying. Slave shelf controllers and slot line cards do not support this command. (Slot host cards do not include the `-d` option.)

```
admin> acct-failsafe
usage: acct-failsafe -option [ params ]
       -d <shelf> <slot>
           (d)isplay AFS info for <shelf> <slot>
       -d (d)isplay AFS info for all relevant slots
       -t (t)oggle module debug level
       -? display this summary
```

To display information about the calls on any slot which are candidates for proxy accounting.:

```
admin> acct-failsafe -d
Slot 1/8:
HashTable @ 10542160, bucketCount: 192, callCount: 23, hashName <afs-1:8>
Slot 2/5:
HashTable @ 10585730, bucketCount: 48, callCount: 7, hashName <afs-2:5>
```

To display the same information for a single slot card in shelf 1, slot 8:

```
admin> acct-failsafe -d 1 8
Slot 1/8:
HashTable @ 10542160, bucketCount: 192, callCount: 23, hashName <afs-1:8>
```

To specify which level of debug to use for the command, use the –t option. A debug level of zero indicates none (no messages). A level of 7 is fairly verbose.

Addrpool

Description: Displays messages related to dynamic address pooling. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`addrpool`** at the command prompt.

Example: Following are several examples of output produced when Addrpool is active.

With 18 addresses currently allocated from a pool:

```
ADDRPOOL: lanAllocate index 0 inuse 18
```

The address 208.147.145.155 was just allocated:

```
ADDRPOOL: allocate local pool address [208.147.145.155]
```

The address 208.147.145.141 is to be freed because the user of that address has hung up. The MAX TNT must find the pool to which the address belonged, then free the address so it is available for another user:

```
ADDRPOOL: found entry by base [208.147.145.141] entry
[208.147.145.129]
ADDRPOOL: free local pool address [208.147.145.141]
```

In the IP Global profile, the Pool-Base-Address [1] is set to 192.168.8.8, and Assign-Count [1] is set to 4:

```
ADDRPOOL: Deleting addrPool
ADDRPOOL: New Addr pool rc = 0
addrPool index 1 ip [192.168.8.8] count 4
```

The Assign-Address parameter of an existing pool is changed from 4 to 3:

```
ADDRPOOL: Deleting addrPool
ADDRPOOL: New Addr pool rc = 0
addrPool index 1 ip [192.168.8.8] count 3
```

A second pool is created. In the IP Global profile, the Pool-Base-Address [2] is set to 192.168.8.8, and Assign-Count [2] is set to 10:

```
ADDRPOOL: Deleting addrPool
ADDRPOOL: New Addr pool rc = 0
addrPool index 1 ip [192.168.8.8] count 4
ADDRPOOL: New Addr pool rc = 0
addrPool index 1 ip [192.168.8.8] count 4
addrPool index 2 ip [192.168.10.1] count 10
```

The second pool is deleted:

```
ADDRPOOL: Deleting addrPool
ADDRPOOL: New Addr pool rc = 0
addrPool index 1 ip [192.168.8.8] count 4
```

ATMPdebug

Description: Displays messages related to Ascend Tunnel Management Protocol (ATMP) sessions. (ATMP is described in RFC 2107.) The command is a toggle that alternately enables and disables the debug display. You would normally use this command with the Tunneldebug command.

Usage: Enter **`atmpdebug`** at the command prompt.

Example:

The mobile node sends a request to foreign agent asking for connection to the home agent:

```
ATMP: sendRegReq: HA=200.67.1.254:5150  RcvUdp=5150
ATMP:    Id=162, FA=130.67.40.254
ATMP:    MC=141.111.40.82, HomeNetName=[]
```

The home agent sets up a tunnel:

```
ATMP: received cmd <RegisterRequest> from 130.67.40.254:5150
ATMP: procRegReq: from=130.67.40.254:5150
ATMP: FA=130.67.40.254, MC=141.111.40.82, HomeNet=
ATMP: sendChallReq: to 130.67.40.254:5150, Id=162, EC=Good completion
ATMP: received cmd <ChallengeReply> from 130.67.40.254:5150
```

```
ATMP: procChallReply: from 130.67.40.254:5150, Id=162
ATMP: sendRegisterReply: to udp=5150, Id=162, Tunnel=156, EC=Good com-
pletion
```

AuthenDebug

Description: Displays messages related to Link Control Protocol (LCP) authentication on the MAX TNT. The command is a toggle that alternately enables and disables the debug display. This command is available on host cards such as the HDLC card and the modem card, and on dual host and network cards such as the SWAN card and the FrameLine card.

Usage: `authendebug`

Example: The following display indicates a successful PAP authentication.

```
AUTH: lcp_pap_req(remote=0)
AUTH-3: verify_pap(given<len.id=13:140.57.40.135, pwdLen=6>)
AUTH-3: verify_pap No authData - getting one
AUTH-3: verify_pap: authDispatcher() == OK
AUTH-3: verify_pap_callback: AUTHCOMMAND_SUCCESS
```

BrouterDebug

Description: Displays messages related to the router functionality of the MAX TNT. The command is a toggle that alternately enables and disables the debug display.

You can use this command for a general view of the load experienced by the MAX TNT.

Usage: Enter **`brouterdebug`** at the command prompt.

Example: Typically, `brouterdebug` displays very few messages. The following session took place over a period of several minutes on a MAX TNT with 40–45 users active.

```
admin> brouterdebug
BROUTER debug display is ON
BROUTER_LOAD_MSG: time= 0
BROUTER_LOAD_MSG: time= 1
BROUTER_LOAD_MSG: time= 0
admin> brouterdebug
BROUTER debug display is OFF
```

The BROUTER_LOAD_MSG message is an indication of how busy the MAX TNT router function is. A low number, as is illustrated here, indicates the router is not experiencing any problems.

BrouterLoad

Description: Reports router backlog time, which indicates whether the MAX TNT is experiencing any delay. The time is shown in ticks. Multiply the number of ticks by ten to get the time in milliseconds.

You can use this command for a general view of the load experienced by the MAX TNT.

Usage: Enter **`brouterload`** at the command prompt.

Example: The following display indicates no delays in the router.

```
admin> brouterload
BROUTER load time is 0 ticks (x10msec)
```

Coredump

Description: Enables or disables the ability of the MAX TNT to send the contents of its memory (core) to a specified UNIX host. The UNIX host must be running the Ascendump daemon, which is available by contacting Ascend Technical Support. For details of using core dumps, see Appendix A, "Getting MAX TNT Core Dumps."

Coredump is a particularly useful tool for Ascend's development engineering, and is occasionally requested by Technical Support for troubleshooting specific issues.

Caution: Using this command will cause the MAX TNT to reboot after its memory (core) has been dumped. Do not use this command unless specifically requested by an Ascend representative.

The Coredump command's syntax provides the following valid entries:

```
coredump
coredump enable | local | remote [server ]
coredump disable
coredump now
coredump trace
```

Syntax element	Description
`coredump`	With no option specifies, reports the enabled or status of Coredump and the core-dump server, if any.
`enable`	Enables Coredump. If you do not specify a server, the core-dump server remains unchanged.
`local`	The most commonly used mode. In Local mode the Ascendump daemon listens for packets from the MAX TNT. The Ascendump daemon operates in server mode, and the MAX TNT core dump facility operates in client mode.
`remote`	Enables the Ascendump daemon to pull a core dump from the MAX TNT. Remotely initiated core dumps can be a security risk, so they are disabled by default. If you enable remote core dumps, they remain enabled only until the MAX TNT resets. That is, a reset restores the default setting.
`server`	The host that has the Ascendump daemon installed. If you do not specify a server, the MAX TNT uses the previously configured server. To specify that the broadcast address be used, enter a hyphen (-).
`disable`	Disables Coredump.
`now`	Forces an immediate core dump to the machine running the Ascendump daemon. This is useful for testing the core dump process.

Syntax element	Description
`coredump`	With no option specifies, reports the enabled or status of Coredump and the core-dump server, if any.
`trace`	Toggles serial debug traces which can be useful to an Ascend representative if a customer is having difficulties.

Example: Following are examples of use of the Coredump command:

```
admin> coredump local 172.31.4.34
coreDump: Sending arp request...
core dump server is '172.31.4.34 ip=[172.31.4.34/16],
mac=[00:60:83:7d:15:8f]
coredump over UDP is enabled locally only with server
172.31.4.34

admin> coredump disable 1.1.1.1
coredump over UDP is disabled locally only with server 1.1.1.1

admin> coredump
coredump over UDP is disabled locally only with server 1.1.1.1

admin> coredump enable 200.200.28.193
coreDump: Sending arp request...
coreDump: Sending arp request...
coreDump: Sending arp request...
coreDump aborted: Can't find ether address for first hop to
200.200.28.193
```

Ctcheck

Description: Analyzes the CIDR tree and displays general statistics about the quantity of nodes and levels in the CIDR tree.

Usage: Enter **`ctcheck`** at the command prompt.

Example:

```
admin> ctcheck
free nodes:   1309
active nodes: 181
total nodes:  492
active%:      36%
max level:    15
ave level:    10.37
```

Ctdebug

Description: Displays messages related to CIDR routing. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`ctdebug`** at the command prompt.

Example:

```
admin> ctdebug
CIDR tree debug is 0
```

Cubit

Description: Displays the statistics of the shelf controller's 3 cubit chips. The command used to gather statistics from the chips. The cubit chips direct packet-bus traffic between each other, between each other and the shelf-controller SAR and slot card SAR, and intershelf to other cubits.

Usage: `cubit -s|-r|-w|-d|-i|-z [ parameters ]`

Syntax element	Description
`-s`	Show the statistics of all 3 cubits.
`-r cubit` *`vpi`*	Read the content of the cubit at the specified VPI.
`-w cubit` *`vpi`* `type fwcubit on|off`	Write to the cubit at the specified VPI. This command is only useful for Ascend development engineers.
`-d cubit` *`startVPI`*	Dump the content of the cubit.
`-i`	Re-initialize the cubits.
`-z`	Reset the cubits.

Example:

In the following example, cubit B is not operating normally and the shelf controller might be defective. Note that receive errors are logged as resets:

```
admin> cubit -s
     Cells     CUBIT_A     CUBIT_B     CUBIT_I
     -----     -------     -------     -------
 received:     1316806           0     2740761
discarded:           0           0           0
misrouted:           0           0           0
HEC Error:           0           0           0
   Resets:           0        5151           0
```

The following example indicates that the multishelf is operating normally:

```
admin> cubit -s

Cells     CUBIT_A     CUBIT_B     CUBIT_I
     -----     -------     -------     -------
 received:      557088     3990584     3497345
discarded:           0           0           0
misrouted:           0           0           0
HEC Error:           0           0           0
   Resets:           0           0           0
```

DTunnel

Description: Displays the status of enabled tunnels on the MAX TNT.

Usage: Enter **`dtunnel`** at the command prompt.

Example:

```
admin> dtunnel
MajDev  Proto  Agent Mode      HA Type  IPX sap   UDP   password
------  -----  ----------      -------  -------  -----  --------
   7     ATMP  Home-Agent      Router   disabled  5150  ascend
   Idle-Limit 120 mins

Tunnels:
------------------------------------------------
Tunnel 36734  IfNum 65535  Majdev 7  Agent Address 130.67.40.254:5150
 Ident=0x56  TN=0x47BF  DnsSN=0
 ATMP Home-Agent
 State 5 (UP)  Router Mode  Home Network Name
 Remote client   Idle-Limit 120 mins
 Client IP Address 141.111.40.86/32
------------------------------------------------
Tunnel 36732  IfNum 65535  Majdev 7  Agent Address 130.67.40.254:5150
 Ident=0x55  TN=0x47BE  DnsSN=0
 ATMP Home-Agent
 State 5 (UP)  Router Mode  Home Network Name
 Remote client   Idle-Limit 120 mins
 Client IP Address 141.111.40.85/32
```

Ether-Stats

Description: Displays all statistics and error counters maintained by the 10Base-T Ethernet driver.

`Usage:` `ether-stats 0 n`

Where `0` is the first Ethernet port for which to display statistics and `n` is the last.

Example:

```
admin> ether-stats 0
Tx unicast:     48382
   non-unicast: 23736
   octets:      10746332
   collisions:  443
   dma under:   0
   cts loss:    0
   no carrier:  0
   late coll:   0
Rx unicast:     45952
   non-unicast: 31307
   octets:      13491043
```

```
collisions:  0
short frame: 0
dma over:    0
no resource: 0
Alignment:   0
Unaligns:    0
Length Errs: 0
Restarts:    0


admin> ether-stats 0-10
Tx unicast:     48559
   non-unicast: 23784
   octets:      10805138
   collisions:  443
   dma under:   0
   cts loss:    0
   no carrier:  0
   late coll:   0
Rx unicast:     46165
   non-unicast: 31500
   octets:      13576590
   collisions:  0
   short frame: 0
   dma over:    0
   no resource: 0
   Alignment:   0
   Unaligns:    0
   Length Errs: 0
   Restarts:    0
```

FRDLstate

Description: Displays information regarding the state of the Frame Relay connections, focusing mostly on Data Link information. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`frdlstate`** at the command prompt.

Example:

```
admin> frdlstate
FRDLCALL state display is now ON
```

In this example, an outgoing call is to be placed. A route to the destination is available over a Frame Relay link. The following message appears:

```
FRDLCALL: Clear Call for route: 136
```

The following message indicates that an outgoing call is connected:

```
FRDLCALL-136: call complete, status 1, 0 channels
```

The next message indicates that either the MAX TNT or the far end device has destroyed a route. The MAX TNT updates its table to reflect this routing change.

```
FRDLCALL-136: dead call
FRDLCALL-136: route destroyed
```

FRdump

Description: Displays a snapshot of the Frame Relay Interface table. The display shows data for each DLCI assigned to a Frame Relay link.

Usage: Enter **`frdump`** at the command prompt.

Example:

```
admin> frdump
* Frname  State  DLinkAddr  routeID.id frmgrLink dlIfNum dlIfSpeed
  frt14    CONNECTED  1012c920  15  0 738 512000
   *dlci Addr  ifNum routeID   dataLink   state
304    100cada0    23   136       1012c920     INACTIVE
   frt18    CONNECTED  1012ffa0  14  0 742 1536000
   *dlci Addr  ifNum routeID   dataLink   state
306    101719a0    33   36       1012ffa0     ACTIVE
 604    10193c60    27   32       1012ffa0     ACTIVE
 603    10191fe0    26   31       1012ffa0     ACTIVE
   frt17    CONNECTED  10149b60  13  0 741 1536000
   *dlci Addr  ifNum routeID   dataLink   state
305    101975e0    32   35       10149b60     ACTIVE
 600    101910a0    24   30       10149b60     ACTIVE
 303    1018cea0    22   28       10149b60     ACTIVE
 301    10186360    20   26       10149b60     ACTIVE
frt16    CONNECTED  1017ad20  7  0 740 1536000
   *dlci Addr  ifNum routeID   dataLink   state
605    101961e0    29   34       1017ad20     ACTIVE
 300    1018a820    21   27       1017ad20     ACTIVE
frswan4    CONNECTED  10125ba0  2  0 734 64000
   *dlci Addr  ifNum routeID   dataLink   state
411    101592a0    31   5       10125ba0     ACTIVE
 407    10155ae0    30   4       10125ba0     ACTIVE
 403    10153be0    25   3       10125ba0     ACTIVE
```

FRinARP

Description: Performs an Inverse ARP test over the specified Frame Relay link and DLCI. You can use FRinARP to help troubleshoot connectivity and routing problems over a Frame Relay link.

Usage: **`frinarp Frame_Relay_profile_name DLCI`**

Example:

```
admin> frinarp FR-1 38
frInArp: frinarp  frname  dlci
Inverse Arp op 2304 hw type 3840 prot type 8 hw len 2 prot len 4
Source Hw address 0401 Target Hw address 0000
Source Protocol address cd933401 Target Protocol address cd930005
```

FRLinkState

Description: Displays Frame Relay control messages. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`frlinkstate`** at the command prompt.

Example:

```
admin> frlinkstate
FR control msg display is ON
```

The following message indicates that the MAX TNT sent a Frame Relay Status Enquiry. The Send sequence number is 135. The Receive sequence number is 134.

```
FRMAIN: time 67192300, send status enquiry (135,134)
```

The next message indicates that DLCI 16 is being processed. This is a normal message. You should see one `process` message for each DLCI.

```
process pvc dlci 16
```

FRLMI

Description: Displays Frame Relay Local Management Interface (LMI) information. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter `frlmi` at the command prompt.

Example:

```
admin> frlmi
FRMAIN: Lmi display is ON
FRMAIN: Setting timer DTE
```

The following message validates the consistency of sequence numbers in LMI messages. The 144 after `want` indicates the original sequence number the MAX TNT sent. The two numbers after the second `got` indicate the switch's Send sequence number and the Switch's report of the last sequence number it received from the MAX TNT, respectively. The original sequence number should match the switch's report of the last sequence number it received.

```
FRMAIN: Time 67201400, got link report: want (*,144), got (144,144)
```

FRMgrDump

Description: Displays the Frame Relay link and DLCI information, including states and counters.

Usage: Enter **`frmgrdump`** at the command prompt.

Example:

```
admin> frmgrdump
Data Link Info
Status
B04FBD40 ACTIVE   B04C0480 1532   19759603   19530429
Status
enq sent =      66710              rsp rcvd =      66763
```

```
upd rcvd =         53               timeouts =          1
Errors
UI field =          0               PD field =          0
CR field =          0               msg type =          0
stat rsp =          0               lock shf =          0
inv info =          0               rpt type =          0
Last Error
type = 5
time =       6100
Fr Type 0      value: 20 octets @ B04FBE26
[0000]: 04 91 03 CC 45 00 00 3A 4B 0E 00 00 7F 11 54 D7
[0010]: CD 93 08 07
LMI type =  AnnexD
DTE Monitor  n391 = 6, t391 = 10, n392 = 3, n393 = 4
Event: recv seq 155 send Seq 155 Index = 0, cycles left = 4
OK OK OK OK OK OK OK OK OK OK
DCE Monitor  t392 = 15,n392 = 3, n393 = 4
Event:  dce send seq 0  index = 0
OK OK OK OK OK OK OK OK OK OK
DLCI info
--addr-- dlci --state- userHndl n201 --check- -pkt xmit- -pkt recv-
B04C09A0    0 ACTIVE          0 1532 NO CHECK      66710      66763
---DE--- --FECN-- --BECN-- -crTime-  chgTime  pending
0        0        0      100      100 FALSE
```

FRPriorityErrors

Description: Reports statistics about Frame Relay priority errors on a host card. All values in its output should be zero. A non-zero value indicates an extreme shortage of memory.

For example:

```
hdlc-1/5> frPriorityErrs
Output:
_sendStatusEnquiryNoMbuf: 0
_mkStatusReplyNoBuf:      0
_mkStatusReplyMbuf:       0
```

FRScert

Description: Toggles between Sprint and Frame Relay Forum LMI checks. The default is the Sprint certification policy. In most cases, the default setting is correct and should not be changed.

Usage: Enter **`frscert`** at the command prompt.

Example:

```
admin> frscert
frSCert is FRFCert
admin> frscert
frSCert is SCert
```

FRstate

Description: Displays messages related to Frame Relay state changes. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **frstate** at the command prompt.

Example: An administrator enables the display, data is received from the Frame Relay interface and processed, and the administrator disables the display.

```
admin> frstate
FRrly state display is ON
FRRLYIF: Calling frifRecv routeId 20
FR1490 dataFrom wan entry state 2
FRRLYIF: Send up stack ifnum 1
FRRLYIF: Calling frifRecv routeId 20
FR1490 dataFrom wan entry state 2
FRRLYIF: Send up stack ifnum 7
FRRLYIF: frIfSend ifNum 1
FR1490 data to  wan entry state 2
FRRLYIF: datatoWan datalink B04C0480

admin> frstate
 FRrly state display is OFF
```

GRE

Description: Displays the MAX TNT Generic Routing Encapsulation (GRE) information. The command has little practical use other than as a tool for developmental engineering.

IFMgr

Description: Displays interface-table entries for the Ethernet interface, toggles the debug display, and marks an interface as enabled or disabled. This command is available on the shelf controller and on host cards such as the Ethernet, modem, HDLC, SWAN, and FrameLine cards. The output differs slightly depending on where the command is executed.

Usage: **ifmgr** [**-d** [***ifnam/ifnum***] | **-t**] [**up**|**down** ***ifnum***|***ifname***]

Syntax element	Description
-d	Display interface table entries.
-d ***ifname/ifnum***	Display details of the specified interface name or number.
-t	Toggle debug display.
up | **down** ***ifnum*** | ***ifname***	Enable or disable the specified interface. These options have the same effect as setting the Enabled parameter in the Ethernet profile, and are subject to the same limitations.

Example: To view the IFMgr usage summary for an Ethernet card in slot 4, first open a session to the card:

```
admin> open 1 4
```

Then you can use the `-d` option to view the interface number and name:

```
ether-1/4> ifmgr -d
if  slot:if  u p ifname   mac addr      local-addr
-------------------------------------------------
000 0:00:000 *   pb0      000000000000  0.0.0.0/32
001 1:17:011 *   ie1-4-1  00c07b6d23f0  11.1.1.1/32
002 1:17:013 *   ie1-4-2  00c07b6d23f1  11.1.2.1/32
003 1:17:015 *   ie1-4-3  00c07b6d23f2  11.1.3.1/32
004 1:17:017 *   ie1-4-4  00c07b6d23f3  11.1.4.1/32
005 1:17:019 *   ie1-4-5  00c07b6d23f4  11.1.5.1/32
<end>
```

The IFMgr `-d` output for an Ethernet card contains the following fields:

Field	Description
`if`	Ethernet interface number.
`slot:if`	Shelf, slot and system-wide interface number. (This interface number is reported by executing the IFMgr command on the shelf controller.)
`u`	Flag indicating whether the interface is up (*) or down (-).
`p`	Flag indicating whether the interface is permanent. A `P` indicates a permanent interface. A hyphen (-) or a blank indicates that it is not. A permanent interface is an interface configured in the command-line interface and stored in MAX TNT NVRAM. All the Ethernet interfaces and the virtual interfaces made for Connection profiles are permanent. Transient interfaces are those the MAX TNT builds from RADIUS, TACACS, or an Answer profile. These interfaces have no interface entry when the connection is down.
`ifname`	Interface name.
`mac addr`	Interface MAC address.
`local-addr`	Interface local address.

Following is an example of disabling an interface:

```
ether-1/4> ifmgr down ie1-4-1
```

The IFMgr `-d` output indicates that the interface is disabled by displaying a hyphen instead of an asterisk in the Up (`u`) column:

```
ether-1/4> ifmgr -d
if  slot:if  u p ifname   mac addr      local-addr
-------------------------------------------------
000 0:00:000 *   pb0      000000000000  0.0.0.0/32
001 1:17:011 -   ie1-4-1  00c07b6d23f0  0.0.0.0/32
002 1:17:013 *   ie1-4-2  00c07b6d23f1  11.1.2.1/32
003 1:17:015 *   ie1-4-3  00c07b6d23f2  11.1.3.1/32
004 1:17:017 *   ie1-4-4  00c07b6d23f3  11.1.4.1/32
```

```
005 1:17:019 *   ie1-4-5  00c07b6d23f4  11.1.5.1/32
<end>
```

Note: The Netstat command also displays a hyphen to indicate a disabled Ethernet interface.

To mark an interface as up, use the `up` option:

```
ether-1/4> ifmgr up ie1-4-1
```

An interface can be administratively disabled by using the IFMgr command or by updating the Ethernet profile, or it can be marked as down by the Ethernet driver when Link-State-Enabled is Yes and Link-State is Down. Therefore, using the `Up` option to the IFMgr command does not necessarily enable the interface. However, it does mark the interface as up.

Following is an example of using the IFMgr command on the shelf controller:

```
admin> ifmgr -d
bif slot sif u m p ifname      host-name  remote-addr       local-addr
-------------------------------------------------------------------------------
000 1:17 000 *     ie0        -          0.0.0.0/32        192.168.7.133/32
001 1:17 001 *     lo0        -          0.0.0.0/32        127.0.0.1/32
002 0:00 000 *     rj0        -          0.0.0.0/32        127.0.0.2/32
003 0:00 000 *     bh0        -          0.0.0.0/32        127.0.0.3/32
004 0:00 000 *     wanabe     -          0.0.0.0/32        127.0.0.3/32
005 0:00 000 *     local      -          0.0.0.0/32        127.0.0.1/32
006 0:00 000 *     mcast      -          0.0.0.0/32        224.0.0.0/32
007 0:00 000 -     tunnel7    -          0.0.0.0/32        192.168.7.133/32
008 1:11 001 *   p wan8       tnt-t1-t32 200.2.1.2/32      192.168.7.133/32
009 1:11 002 *   p wan9       tnt-t1-t32 200.2.2.2/32      192.168.7.133/32
010 1:11 003 *   p wan10      tnt-e1-t22 200.3.2.2/32      192.168.7.133/32
011 1:11 004 *   p wan11      tnt-e1-t32 200.5.1.2/32      192.168.7.133/32
012 1:11 005 *   p wan12      tnt-e1-t32 200.5.2.2/32      192.168.7.133/32
013 1:11 006 *   p wan13      tnt-t1-t22 200.1.1.2/32      192.168.7.133/32
014 1:15 001 *   p wan14      tnt-t1-s1- 100.1.100.2/32    100.6.100.2/32
015 1:11 007 *   p wan15      tnt-e1-t22 200.3.1.2/32      192.168.7.133/32
016 1:11 008 *   p wan16      cisco-t221 200.4.103.2/32    192.168.7.133/32
017 1:11 009 *   p wan17      m-e1-t2211 200.4.4.2/32      192.168.7.133/32
018 1:11 010 *   p wan18      m-e1-t2212 200.4.4.3/32      192.168.7.133/32
019 1:17 000 -   p wan19      m2t81      200.8.1.2/32      192.168.7.133/32
020 1:17 000 -   p wan20      m41        200.4.1.2/32      200.6.1.2/32
021 1:16 001 *   p wan21      p1321n<>p1 0.0.0.0/32        0.0.0.0/32
[More? <ret>=next entry, <sp>=next page, <^C>=abort]
```

The IFMgr output on cards other than the Ethernet card includes the following fields:

Field	Description
`bif`	Bundle interface number. There is one interface number per bundle, including MPP connections. It is the global interface-table number.
`slot`	Shelf and slot the interface is assigned to.
`sif`	Slot interface.
`u`	Flag indicating whether the interface is up (*) or down (-).

Field	Description
m	Indicates that the interface is part of an MP bundle.
p	Flag indicating whether the interface is permanent. A P indicates a permanent interface. A hyphen (-) or a blank indicates that it is not. A permanent interface is an interface that is configured in the command-line interface and stored in MAX TNT NVRAM. All the Ethernet interfaces and the interfaces based on Connection profiles are permanent. Transient interfaces are those the MAX TNT builds from RADIUS, TACACS, or an Answer profile. These interfaces have no interface entry when the connection is down.
ifname	Interface name.
host-name	Host name of remote device.
remote-addr	Remote address of device as configured in a Connection profile.
local-addr	Local address of device as configured in a Connection profile.

Following is an example of displaying information about a particular interface:

```
admin> ifmgr -d 009
inUse:          Yes
hostName:       tnt-t1-t3212-s4
dialoutName:
ExternalAuth:   No
ExternFilters:  No
ExternRoutes @  0
ExternIpxRoutes @       0
miscInfo @      0
reDirectDest:   0.0.0.0
DLCI routeId:   34
MP(P) id:       0
Logical iff:    2
virtual id: 0, virtual next @ 0, virtual main @ 0
minor device:   9
device status:  0x303
mtu:            1528
ip_addr:        192.168.9.133
dstip_addr:     100.2.1.2
netmask:        255.255.255.0
net:            192.168.9.0
subnet:         192.168.9.133
bcast:          192.168.9.255
nbcast:         192.168.9.133
directed-bcast:         no
macaddr:        000000000000
inp_qcnt:       0
out_qcnt:       0
nexthop:        0.0.0.0
Num pkts queued for brouter:    0
proxy_arp_mode: 0
```

```
proxy_arp_head: 0
No associated connection profile
```

The ICMP-Reply-Directed-Bcast parameter in the IP-Global profile specifies whether the MAX TNT responds to directed-broadcast ICMP echo requests. If set to No, the system does not respond to any directed-broadcast ICMP requests. The setting of this parameter is shown in the Directed-Bcast field in the Ifmgr output.

IPXRIPdebug

Description: Displays incoming and outgoing IPX RIP traffic. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`ipxripdebug`** at the command prompt.

Example:

```
admin> ipxripdebug
IPX-RIP state display is ON
```

The following message appears as the MAX TNT sends an IPX RIP packet announcing its route:

```
IPXRIP: 10000a17 announced 0 routes on interface 1000:
```

Next, a Pipeline 50 has dialed the MAX TNT. The MAX TNT receives a RIP route from the Pipeline.

```
IPXRIP: received response from ac1b0001:00c07b5e04c0 (1 nets).
```

The following message indicates that the MAX TNT is delaying sending a RIP packet to prevent the interpacket arrival time from being shorter than busy/slow boxes can handle. An IPX router should never violate the minimum broadcast delay.

```
IPX-RIP: too soon to send on interface 1000.
IPXRIP: 10000a81 announced 0 routes on interface 1000:
IPXRIP: received response from ac1b0001:00c07b6204c0 (1 nets).
IPXRIP: 10000aa6 announced 0 routes on interface 1000:
IPXRIP: received response from ac1b0001:00c07b5504c0 (1 nets).
IPXRIP: 10000abc announced 0 routes on interface 1000:
```

Lanval

Description: Displays messages related to external validation requests. You can use this command in conjunction with `radif` to troubleshoot authentication issues.

Usage: Enter **`lanval`** at the command prompt.

Example:

```
admin> lanval
LANVAL state display is ON
LANVAL: radius auth, id B054AD60
LANVAL: radius callback, id B054AD60, auth SUCCESS
```

```
LANVAL:_lanvFreeInfo: freeing iprof@B05A9360
```

LifDebug

Description: Displays ISDN layer 2 and layer 3 information. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`lifdebug`** at the command prompt.

Example: Following are several examples of `LifDebug` output:

```
admin> lifdebug
LIF debug is now ON
```

A packet is being sent over the WAN. The packet is constructed:

```
LIF_SendPkt(): DSL 0, source 0x400, destination 0x300,
event 0x340, SAPI 0, CES 1, Call_Id 77, Chan_Id 0
```

The following message displays the contents of the packet:

```
PACKET:
Header (4): a0 50 59 b0  Info (9): 08 02 00 00 84 08 02 80 90 01
L3_Go: source 0x400, event 0x340, DSL 0, call_id 77, ces 1
L3_ProcessUserEvent(): State 0x9, Event 0x84, Index 6,
DSL 0, CallID 77
```

Another packet is sent:

```
LIF_SendPkt(): DSL 0, source 0x300, destination 0x205,
event 0x240, SAPI 0, CES 1, Call_Id 77, Chan_Id 0
PACKET:
Header (4): a0 50 59 b0  Info (9): 08 02 83 fe 45 08 02 80 90 00
L3_Go(): end of L3 task, NLCB State 10
L2_Go(): DSL_Id=0, SAPI=0, CES=1, TEI=0, Event=240
L2_ProcessEvent(): DSL 0, index 13, state 7
L2_ProcessEvent(): DSL 0, index 19, state 7
L2_Go(): DSL_Id=0, SAPI=0, CES=1, TEI=0, Event=1
L2_ProcessEvent(): DSL 0, index 1, state 7
L2_Go() end: DLCB->State 7
```

MdbStr

Description: Modifies the default modem AT command strings used by the modems on the MAX TNT for both incoming and for outgoing calls. Previously, you could not modify the AT command for modems on the MAX TNT. You could only affect the string in minor ways by modifying the parameters in the Terminal-Server>Modem-Configuration subprofile. Note that when the modem card or the MAX TNT is reset, the AT command strings revert to their defaults.

The MdbStr command also allows you to return the string to its factory default settings.

The modem chip in the MAX TNT supports AT commands up to 56 characters in length. To fully support all possible functionality, each command is sent as two separate strings. You can modify one or both strings.

Caution: The AT command string initializes the modems it supports. When you change the AT command string, you are changing the functionality of the modems. Use this command with caution.

Here are the two default strings for the MAX TNT:

1 `AT&F0&C1V0W1X4`

2 `AT%C3\N3S2=255S95=44S91=10+MS=11,1,300,33600A`

Usage: `mdbstr [ 0 ] [ 1 ] [ 2 ] [ AT-command-string ]`

Example: The following examples show you how to modify each portion of the AT command string:

To override the existing first string with a new string:

```
mdbstr 1 AT&F0&C1V1W1
```

This will override the second portion of the AT command string:

```
mdbstr 2 AT%C3\N3S2=255S95=44S91=10+MS=11,1,300,14400A
```

This will return both strings to their factory default settings:

```
mdbstr 0
```

MDialout

Description: Displays messages related to modem dial out. This command can used in conjunction with the ModemDrvState command to get detailed information about outbound modem calls.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`mdialout`** at the command prompt.

Example: In the following example, a modem on the MAX TNT prepares to make an outbound modem call, but never receives a dialtone.

```
admin> mdialout
MDIALOUT-2/4: >> CURR state=Await_Off_Hook, NEW
event=Event_Off_Hook
MDIALOUT-2/4: connected to DSP!
MDIALOUT-2/4: rqst tone (14) via channelIndex 0
MDIALOUT-2/4: tone generation started.
MDIALOUT-2/4: >> CURR state=Await_Dial_Tone, NEW
event=Event_Dialtone_On
MDIALOUT-2/4: decode timer started.
MDIALOUT-2/4: << NEW state=Await_1st_Digit
MDIALOUT-2/4: enabling tone search, channel index=0,
timeslot=0
MDIALOUT-2/4: << NEW state=Await_1st_Digit
```

```
MDIALOUT-2/4: >> CURR state=Await_1st_Digit, NEW
event=Event_On_Hook
MDIALOUT-2/4: stopping decode timer.
MDIALOUT-2/4: rqst tone (15) via channelIndex 0
MDIALOUT-2/4: disabling tone search, channel index=0
MDIALOUT-2/4: disconnected from DSP.
MDIALOUT-2/4: << NEW state=Await_Off_Hook
MDIALOUT-2/4: >> CURR state=Await_Off_Hook, NEW
event=Event_Close_Rqst
MDIALOUT-?/?: << NEW state= <DELETED>
```

MDialSess

Description: Displays all the active modem dialout sessions.

Usage: Enter `mdialsess` at the command prompt.

Example:

```
admin> mdialsess
entry slot:mdm route port hookDetect DSP:tone:timr:decode state
1    6:4     145   16 pollForOff  n : n  : n  :  n    Await_Off_Hook
```

ModemD1Stats, ModemD2Stats, ModemD3Stats

Description: Displays modem statistics. ModemD1Stats displays statistics for the first 16 modems, ModemD2Stats displays statistics for the second 16 modems, and ModemD3Stats displays statistics for the last 16 modems.

Usage: `modemd1stats`

To use this command, first open a session with a modem card, then enter the command.

Example:

```
modem-1/2> modemd1stats
modem:  ansFail    ansOK   1-2400 2.4-14.4  14.4-up  21.6+up  28.8+up
 1/ 0:        3      171        0        0     171      171      171
 1/ 1:        3      171        0        0     171      171      171
 1/ 2:        2      172        0        0     172      172      172
 1/ 3:        2      172        0        0     172      172      171
 1/ 4:        4      170        0        0     170      170      170
 1/ 5:        1      173        0        0     173      173      172
 1/ 6:        0      174        0        0     174      174      174
 1/ 7:        1      173        0        0     173      173      173
 1/ 8:        1      173        0        0     173      173      173
 1/ 9:        0      174        0        0     174      174      174
 1/10:        2      172        0        0     172      172      172
 1/11:        1      173        0        0     173      173      173
 1/12:        1      173        0        0     173      173      173
 1/13:        0      174        0        0     174      174      174
 1/14:        1      173        0        0     173      173      173
```

```
 1/15:        3      171        0        0     171      171      170
```

ModemDrvDump

Description: Displays information about the status of each modem.

Usage: Enter **`modemdrvdump`** at the command prompt.

Example: Following is a message about modem 0 (the first modem) in the modem card in slot 3 on the MAX TNT. The numbers in brackets indicate number of calls with unexpected open requests, unexpected Rcode events, unexpected release events, and unexpected timeouts:

```
MODEMDRV-3/0: Unexp Open/Rcode/Rlsd/TimOut=[0,0,0,0]
```

ModemDrvState

Description: Displays communication to and from the modem driver on the MAX TNT. You can see which buffers are allocated and which AT command strings are being used to establish modem connections.

You can also determine whether data is received from the modem in an understandable format. If line quality is poor, the modem driver attempts to parse incoming data from the modem, but it might not be successful. This command can used in conjunction with the MDialout command to get detailed information about outbound modem calls.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`modemdrvstate`** at the command prompt.

Example: Following are examples of a modem call coming into the MAX TNT, and a modem call being cleared from the MAX TNT:

```
admin> modemdrvstate
MODEMDRV debug display is ON
```

Modem 1 on the modem card in slot 3 has been assigned to answer an incoming modem call:

```
MODEMDRV-3/1: modemOpen modemHandle B04E3898, hdlcHandle B026809C,
orig 0
```

The modem is idle, so it is available to answer the call:

```
MODEMDRV-3/1: _processOpen/IDLE
```

The next two lines show the MAX TNT modem sending the first string:

```
MODEMDRV: Answer String, Part 1 - AT&F0E0+A8E=,,,0
```

A buffer needs to be allocated for sending the command out to the WAN:

```
MODEMDRV-3/1: _hdlcBufSentFnc: buffer = 2E12EAE0, status = SENT
```

Buffers are allocated for data being received from the WAN:

```
MODEMDRV-3/1: _hdlcBufRcvdFnc: data=2E13ADF0, len=8, pars-
eState[n,v]=[0,0], status= RCVD
```

```
MODEMDRV-3/1: _hdlcBufRcvdFnc: data=2E13BA20, len=5, pars-
eState[n,v]=[0,0], status= RCVD
```

The MAX TNT modem receives an OK from the calling modem:

```
MODEMDRV-3/1: data =OK
```

The process is repeated for strings 2 and 3:

```
MODEMDRV-3/1: processTimeout/DIAL_STR2[2D]
MODEMDRV: Answer String, Part 2 - AT&C1V1\V1W1X4S10=60
MODEMDRV-3/1: _hdlcBufSentFnc: buffer = 2E12EAE0, status = SENT
MODEMDRV-3/1: _hdlcBufRcvdFnc: data=2E13C038, len=2, pars-
eState[n,v]=[0,0], status= RCVD
MODEMDRV-3/1: data = 0
MODEMDRV-3/1: _processTimeout/DIAL_STR3
MODEMDRV: Answer String, Part 3 -
AT%C3\N3S2=255S95=44S91=10+MS=11,1,300,33600,A
```

Now result codes are processed to clarify the characteristics of the connection.

```
MODEMDRV-1/1: _hdlcBufRcvdFnc: data=9880C628, len=48, pars-
eState[n,v]=[1,0], stD
MODEMDRV-1/1: data =
CONNECT 115200/V34/LAPM/V42BIS/28800:TX/33600:
MODEMDRV-1/1: decodeSLC[15]=<CONNECT 115200/> checking for error cor-
rection
MODEMDRV-1/1: decodeSLC[4]=<V34/> checking for error correction
MODEMDRV-1/1: decodeSLC[5]=<LAPM/> checking for error correction[29]
MODEMDRV-1/1: decodeSLC[7]=<V42BIS/> checking for compression[21]
MODEMDRV-1/1: decodeSLC[9]=<28800:TX/> checking for xmit[1]
MODEMDRV-1/1: _hdlcBufRcvdFnc: data=9880C828, len=4, pars-
eState[n,v]=[4,0], staD
MODEMDRV-1/1: data = RX
> checking for recv[0]C[9]=<33600:RX
decodeSLC complete
```

At this point the modem call is up, and the modem driver has completed its tasks. The call will be passed to Ethernet resources:

```
MODEMDRV-3/1: _processRcodeEvent/AWAITING RLSD, mType=5, RLSD=0
MODEMDRV-3/1: _processRlsdChange/AWAITING RLSD = 1
```

Following is the normal sequence of steps for a modem call that is cleared (by either modem). Modem 5 on the modem card in slot 7 of the MAX TNT is freed from the previous call, and it is reinitialized (so it is available for the next call).

```
MODEMDRV-7/5: modemClose modemHandle B04E6F38
MODEMDRV-7/5: _closeConnection:ONLINE, event=3
MODEMDRV-7/5: _processTimeout/INIT
```

MPCMtoggle

Description: Displays information about related channel addition with Multilink Point-to-Point connections. This information is not related to MP+ or BACP connections. This

command displays only information from connections established as MP (RFC1717) connections.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`mpcmtoggle`** at the command prompt.

Example:

```
admin> mpcmtoggle
MPCM debug is now ON
MPCM-432: adding 1 channels
```

MPentry

Description: Displays information for a specified, active, MP or MP+ connection, including the options negotiated the connection. This command can be extremely helpful when researching MP or MP+ compatibility issues.

Note: The MpID number that must be entered is an internally generated number. To get a list of all currently assigned MpID numbers on your MAX TNT, enter the `IFmgr -d` command and specify a interface name or number.

Usage: Enter **`mpentry`** at the command prompt.

Example: The following example shows an MP+ call (noted as MPP). The End Point Discriminator (used to bundle the channels together) is shown under `bundle id`. In this case, it is the hardware MAC address of the calling device.

```
admin> mpentry
MpID required
admin> mpentry 28
MP entry 28 @ B055DE60
MpID  28, Flags: delete No, remote No, ncp Yes, mpp Yes bacp No
bundle id: 15 octets @ B0558BE0
[0000]: 03 00 C0 7B 53 97 07 73 65 63 61 2D 68 73 76
vjInfo @ B0562060
startTime 227521989, mrru: local 1524, peer 1524
send: ifIx 1, count 0, seq 77268 / recv: seq 75046
IF 50, send idle 0, recv idle 1, last seq 75045 mode 0 #chans 1
Head:
Tail
Reassembe packet cnt 0 bad lrg pkts 0
```

MPPCM

Description: Displays MP+ call-management information. The command is a toggle that alternately enables and disables the debug display. You can use it in conjunction with the MPtoggle command, since each command logs debug from a different place in code, but both display information based on multichannel connections.

Usage: Enter **`mppcm`** at the command prompt.

Example:

```
admin> mppcm
MPPCM debug is now ON
```

The following 8 messages indicate that a second channel is added to a 1-channel MP+ connection:

```
MPP-5: Event = Utilization, CurrentState = Idle/A
MPP-5: check dynamic says: current = 1, recommended = 2
MPP-5: requesting 1 additional channel(s)
MPP-5: 1 call(s) posssible.
MPP-5: new state is: Add/C
MPP-5: Event = RxAddComplete, CurrentState = Add/C
MPP-5: enterIdleA, AddLock = Yes, RemoveLock = No
MPP-5: new state is: Idle/A
```

The following 12 messages indicate that a remote management session is brought up for the MP+ user with MpID 28. You can open a remote session to an MP+ user from the terminal server.

```
MPP-28: Event = StartRM, CurrentState = Idle/A
MPP-28: start remote management
MPP-28: new state is: Idle/A
MPP-28: Event = RxRmRsp, CurrentState = Idle/A
MPP-28: remote management response (0)
MPP-28: new state is: Idle/A
MPP-28: Event = RxRmTxReq, CurrentState = Idle/A
MPP-28: new state is: Idle/A
MPP-28: Event = RecvRMM, CurrentState = Idle/A
MPP-28: new state is: Idle/A
MPP-28: Event = StopRM, CurrentState = Idle/A
MPP-28: stop remote management
```

```
admin> mppcm
MPPCM debug is now OFF
```

MPtoggle

Description: Displays information about MP and MP+ connections. You can use this command in conjunction with the MPPCM command, since each command logs debug from a different place in code, but both display information based on multichannel connections. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`mptoggle`** at the command prompt.

Example:

```
admin> mptoggle
MP debug is now ON
MP-26: sending control message 191
MP-5: sending control message 76
```

```
admin> mptoggle
MP debug is now OFF
```

MSstat

Description: The MSstat command displays information about communications with other shelves over the intershelf TDM bus. On the master shelf, the command displays statistics for the slave shelves. On a slave shelf, it displays statistics for the master and other slaves.

Usage: msstat

Example: From a master shelf:

```
admin> msstat
SH  State   TxQs  TxSeq  RxSeq Resend  Timer LinkUp
 2      4      0  13312  13116      3      0      2
 3      4      0  12405  11822      0      0      2
 4      2      0      0      0      0      1      0
 5      2      0      0      0      0      1      0
 6      2      0      0      0      0      1      0
 7      2      0      0      0      0      1      0
 8      2      0      0      0      0      1      0
 9      2      0      0      0      0      1      0
```

Note that there are entries for all shelves (2-9), even though this system has only three shelves.

The MSstat command's output includes the following fields:

Field	Description
SH	Shelf number
State	Indicates the state of the multishelf communications. Values can be: • 1—No communications. • 2—Communications are initializing. • 3—Communications are initializing. • 4—Operational.
TxQs	Number of messages in queue but not yet sent.
TxSeq	Number of messages sent.
RxSeq	Number of messages received.
Resend:	Number of retransmitted messages.
Timer	Number of seconds the shelf has been in the current state.
LinkUp	Number of times communications between the shelves have been established.

Example: From a slave shelf:

```
shelf-router-3/17> msstat
SH  State   TxQs  TxSeq  RxSeq Resend  Timer LinkUp
1      4      0  13693  13991      0      0      1
```

Note that on a slave shelf, only the master shelf is shown.

NetIF

Description: Displays the MAX TNT network interface mappings.

Usage: **`netif -m -q -t -v -?`**

Syntax element	Description
`-m`	Display mappings for the specified map type.
`-q`	Display the queue for a map.
`-t`	Toggle debug display.
`-v`	Display valid mapping tables.
`-?`	Display this summary.

Example:

```
admin> netif -v
map 0x1042C0E0: type 0 (call-id), id 0x1042B5A0


admin> netif -m 0
SHELF  SLOT  SysID      SlotID
   1         1      52      2
   1         6      90     58
   1         6      89     57
   1         6      86     56
   1         6      78     51
   1         6      72     50
   1         6      71     49
   1         6      70     48
   1         6      69     47
   1         6      68     46
   1         6      62     45
   1         6      61     44
   .
   .
   .
```

Networki

Description: Displays information about calls as they are first presented to the MAX TNT. The MAX TNT assigns a numeric tag to each call in order to monitor the connection. (After a call passes through this section of code, it typically moves to a call-route manager, and is monitored with the RoutMgr diagnostic command.)

The Networki command is a toggle that alternately enables and disables the debug display.

Usage: Enter `networki` at the command prompt.

Example:

```
admin> networki
NETWORKI debug is now ON
```

The following messages indicate a new call coming into the MAX TNT. This is a normal string of messages for most calls:

```
** CALL 30  RINGING      globDsl  0, channel 23, session 999
```

The call is assigned a callID of 23 and a routeID of 123:

```
NETWORKI: cached callID 30, routeID 123
```

Resources have been allocated for the call. The MAX TNT then begins the process of answering the call:

```
NETWORKI: answering incoming call for route 123
```

The MAX TNT checks its call-route table to verify that it has an entry for the new call:

```
NETWORKI: found callID 30 for route
NETWORKI: found session for route 123
NETWORKI: clearSessionData
NETWORKI: answerCallRequest( 30, 123 )
```

With the next message, the call has been answered. The MAX TNT then determines where to route the call.

```
** CALL 30  CONNECTED    globDsl  0, channel 23, session  26
NETWORKI: call state connected, callid: 30
networki::pending call, checking for session completeness
NETWORKI: completeTransaction, route 123
NETWORKI: First call completed.  Got base profile, service 1, type 0
NETWORKI: activateChannelList for route 123
clear enter NETWORKI: clearSessionData
clear exit clear done
```

At this point, the call is passed to another function, and Networki no longer applies to this call.

Following is a normal string of messages showing a call being cleared.

```
NETWORKI: clearSession
NETWORKI: Aborting transaction, route 102
NETWORKI: clearing retries
NETWORKI: callid 6 added to pending clear list
** CALL 6   INACTIVE     globDsl  0, channel 23, session 999
```

OSPFAVLtree

Description: Displays the entire OSPF AVL tree.

Caution: With earlier software, this command caused the MAX TNT to reset. Do not use this command unless your MAX TNT is running 2.0 or later software.

Usage: Enter **`ospfavltree`** at the command prompt.

Example:

```
admin> ospfavltree
dest            mask           Lptr           Rptr           Myaddr  mrkedDl
0x       0      0x       0     0x       0     0x       0     0x1038dc00 0x0
0x    650a      0x    ffff     0x       0     0x       0     0x104ef168 0x0
0x 200650a      0xffffffff     0x104ef168     0x       0     0x104ef0b8 0x0
0x    660a      0x    ffff     0x104ef0b8     0x104ed66c     0x104ef0e4 0x0
0x 200660a      0xffffffff     0x       0     0x       0     0x104ef13c 0x0
0x    670a      0x    ffff     0x104ef13c     0x104edb68     0x104ed66c 0x0
.
.
.
```

PBecho

Description: Tests the multishelf packet bus by using it to send traffic from one shelf controller to another.

The PBecho command is similar to Ping in that it sends a packet to a known destination and echoes the packet back. Because each cell contains a unique destination address to a shelf and slot within the system, you can test the packet bus by simply sending packets across it.

Usage: `pbecho shelf slot count size`

Syntax element	Description
`shelf`	Specifies the shelf to which to direct the echo packet.
`slot`	Specifies slot to which to direct to the echo packet.
`count`	Specifies the number of packets to send.
`size`	Specifies the size of the packets to send.

Example: In the following example, an administrator sends a thousand 1500-byte packets from the master shelf-controller to the slave shelf-controller in shelf 3:

```
admin> pbecho 3 17 1000 1500
pbus: Echo packets sent: 1000 rcvd: 1000 error: 0
```

The output of the command indicates that the slave controller received 1000 packets and echoed them back to the master shelf-controller. To further test the packet bus traffic across the multishelf system, the administrator could repeat the command a few times with different packet sizes, then, use the same command to send packets from the slave shelf-controller to the master.

PermConn-List

Description: Displays a list of all permanent connection profiles in the MAX TNT.

Usage: Enter **`permconn-list`** at the command prompt.

Pools

Description: Displays a snapshot of a large selection of memory pools, the size of each pool, and the status of each pool. At the end of the list is a summary of the total memory allocation in the MAX TNT.

Memory is dynamically allocated to support various tasks, and should be freed when a particular task has been completed. Taking pools snapshots over an extended period of time can help troubleshoot a problem with a memory leak, in which memory is allocated for a task but never freed.

Snapshots should never show the entire quantity of allocated memory (or even any single pool) increasing over an extended period of time.

Usage: Enter **`pools`** at the command prompt.

Example: The number of pools displayed is usually very large. The following example displays just a portion of the typical output.

```
admin> pools
Pool Name                          size  limit  inUse  hiWat  heapAdrs
Accounting Session Change Registrants      8     0       1      1
103CCAE0
AcctEvnt                            14     0     127    127   103CCAE0
AfsHashEntry                       191     0       0      0   103CCBE0
AfsTaskMsg                         219     0       0      0   103CCBE0
AssignedChannelPool                 32     0     127    139   103CCAE0
AuthData                           116     0       0      0   103CCBE0
BrouterPool                         80     0       2     14   103CCB60
.
.
.
volatile profile instance           16     0     171    184   103CCAE0
volatile profile type info          12     0       7      7   103CCAE0
```

The first portion of the Pools command output includes the following fields:

Field	Description
`Pool name`	Pool name.
`Size`	Size of the pool, in kilobytes.
`Limit`	Maximum number of buffers that can be allocated to a pool.
`InUse`	Number of pools in use.
`HiWat`	Highest number of pools allocated to a task since the MAX TNT was brought up.
`HeapAdrs`	Memory address of pool.

Following the list of pools, the Pools command displays a summary of memory usage:

```
total pools:          175
```

```
          total buffers in use:        10593
                total memalloc:       261685
                 total memfree:       258558
              memalloc in use:         3129
            memalloc failures:            0
             memfree failures:            0
           memalloc high water:         3146
Histogram of memalloc'd memory block sizes:
    2659 buffers in range [64,127]
    632 buffers in range [128,255]
    2 buffers in range [256,511]
    22 buffers in range [512,1023]
    9 buffers in range [1024,2047]
    21 buffers in range [2048,4095]
    3 buffers in range [4096,8191]
    7 buffers in range [8192,16383]
    6 buffers in range [32768,65535]
    2 buffers in range [131072,262143]
    1 buffers in range [262144,524287]
Total memory in use: 1295104 bytes in 3364 buffers
Histogram of free memory block sizes:
    12 buffers in range [128,255]
    1 buffers in range [256,511]
    2 buffers in range [1024,2047]
    1 buffers in range [1048576,2097151]
Total free memory: 1503680 bytes in 16 buffers
```

Following are descriptions of some of the more important fields in this display:

Field	Description
`total pools`	Total number of pools in use.
`total buffers in use`	Number of buffers in use.
`total memalloc`	Total number of times the MAX TNT allocated a block of memory for use.
`total memfree`	Total number of times the MAX TNT freed a block of memory. This should be fairly close to total memalloc.
`memalloc in use`	Total number of memory pools in use. This is the difference between total allocated and total freed.
`memalloc failures`	Total number of times the MAX TNT failed to allocate a block of memory for use.
`memfree failures`	Total number of times the MAX TNT failed to free a block of memory.
`memalloc high water`	The highest number of memory pools in use at any one time.

PortInfo

Description: Displays information about the MAX TNT ports.

Usage: **`portinfo port-number`**

Example:

```
admin> portinfo 1
Printing fixed/allocated ports for slot 1
Linear Port:   1
- fixed:       TRUE
- relative #:  0
- paired port: 65535
- slave:       FALSE
- physical:    FALSE
```

PPPdump

Description: Very similar to the WANdisplay diagnostic command. But the PPPdump command strips out escape characters that are present for asynchronous PPP users (who are dialing in with modems). The escape characters are necessary because of the asynchronous nature of the data stream. Stripping them out simply clarifies the presentation of the data.

If you enter the command while traffic through your MAX TNT is heavy, the resulting amount of output can make it tedious to find the information you're looking for. The screen might even display the message `----- data lost -----`, which just means that not all the output can be displayed on the screen.

You might prefer to use the PPPdump command during a period of low throughput.

Usage: First open a session with a host card, then enter `pppdump` *n*

where **n** is the number of octets to display per frame. Specifying a value of 0 (zero) disables the logging of this data.

Example: Following are two examples of the display of an asynchronous call, one produced by WANdisplay and the other by PPPdump.

The following frames were logged by entering **`wandisplay 64`**:

```
7E FF 7D 23 C0 21 7D 21 7D 21 7D 20 7D 37 7D 22 7D 26 7D 20 7D 2A 7D
20 7D 20 2D 7D 23 7D 26 3A AA 7E
7E FF 7D 23 C0 21 7D 21 7D 21 7D 20 23 7D 20 7D 24 7D 20 7D 20 7D 22
7D 7E
```

To get the data stream without escape characters, the 0x7D bytes need to be stripped, and the byte following each 0x7D byte needs to be decremented by 0x20.

With PPP dump, the data is automatically converted and displayed:

```
7E FF 03 C0 21 01 01 00 17 02 06 00 0A 00 00 2D 03 06 3A AA 7E 7E
FF 03 C0 21 01 01 00 23 00 24 00 00 02 7E
```

See Also: WANdisplay, WANnext, WANopen

PPPFSM

Description: Displays changes to the PPP state machine as PPP users connect. The command is a toggle that alternately enables and disables the debug display.

Usage: First open a session with a host card, then enter **`pppfsm`** at the command prompt.

Example: The following display shows the complete establishment of a PPP session:

```
admin> pppfsm
PPPFSM state display is ON
PPPFSM-97: Layer 0   State INITIAL     Event OPEN...
PPPFSM-97: ...New State STARTING
PPPFSM-97: Layer 0   State STARTING    Event UP...
PPPFSM-97: ...New State REQSENT
PPPFSM-97: Layer 1   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 2   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 3   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 4   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 5   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 6   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 7   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 8   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 9   State INITIAL     Event UP...
PPPFSM-97: ...New State CLOSED
PPPFSM-97: Layer 0   State REQSENT     Event RCONFREJ...
PPPFSM: irc_new scr 4
PPPFSM-97: ...New State REQSENT
PPPFSM-97: Layer 0   State REQSENT     Event RCONFACK...
PPPFSM-97: ...New State ACKRECD
PPPFSM-97: Layer 0   State ACKRECD     Event RCONFREQ...
PPPFSM-97: ...New State ACKRECD
PPPFSM-97: Layer 0   State ACKRECD     Event RCONFREQ...
PPPFSM-97: Layer 1   State CLOSED      Event OPEN...
PPPFSM-97: ...New State REQSENT
PPPFSM-97: ...New State OPENED
PPPFSM: PAP Packet
PPPFSM-97: Layer 6   State CLOSED      Event OPEN...
PPPFSM-97: ...New State REQSENT
PPPFSM-97: Layer 4   State CLOSED      Event OPEN...
PPPFSM-97: ...New State REQSENT
PPPFSM-97: Layer 4   State REQSENT     Event RCONFREQ...
PPPFSM-97: ...New State REQSENT
PPPFSM: ccp Packet code 1
PPPFSM-97: Layer 6   State REQSENT     Event RCONFREQ...
PPPFSM-97: ...New State REQSENT
PPPFSM: ccp Packet code 2
PPPFSM-97: Layer 6   State REQSENT     Event RCONFACK...
PPPFSM-97: ...New State ACKRECD
```

```
PPPFSM-97: Layer 4   State REQSENT     Event RCONFACK...
PPPFSM-97: ...New State ACKRECD
```

PPPinfo

Description: Displays information about established PPP sessions. The command has little practical use other than as a tool for developmental engineering.

Usage: **`pppinfo`** ***`index`*** [**`all`**]

Syntax element	Description
`index`	Selects a particular PPP information table.
`all`	Displays information about embedded structures.

Example:

```
admin> pppinfo 1
Ncp[LCP]        = B02B396C
Ncp[AUTH]       = B02B39BC
Ncp[CHAP]       = B02B3A0C
Ncp[LQM]        = B02B3A5C
Ncp[IPNCP]      = B02B3AAC
Ncp[BNCP]       = B02B3AFC
Ncp[CCP]        = B02B3B4C
Ncp[IPXNCP]     = B02B3B9C
Ncp[ATNCP]      = B02B3BEC
Ncp[UNKNOWN]    = B02B3C3C
Mode            = async
nOpen pending   = 0
LocalAsyncMap   = 0
RemoteAsyncMap  = 0
Peer Name       = N/A
Rmt Auth State  = RMT_NONE
aibuf           = 0
ipcp            = B03E502C
vJinfo          = 0
localVjInfo     = 0
bncpInfo        = B03E559C
ipxInfo         = B03E55DC
remote          = no
Bad FCS         = a
```

PPPstate

Description: Displays the state of a PPP connection. Different PPP calls can be routed (call routing, as opposed to IP or IPX routing) through a MAX TNT differently. The command is a toggle that alternately enables and disables the debug display.

The command has little practical use other than as a tool for developmental engineering.

Usage: Enter **`pppstate`** at the command prompt.

Example: The following message indicates that data is moved directly from the WAN to the Ethernet segment. WAN data can be redirected to other resources (X.75 handler or V.120 handler) before it is ready to be sent to the Ethernet segment.

```
PPP-116: Redirect async wan direct
```

PRIdisplay

Description: Display all ISDN PRI D-channel signaling packets that are either received or sent through the PRI interfaces.

Usage: To use this command, first open a session with a network card configured for PRI signaling (for example, a T1 or E1 card). Then enter the PRIdisplay command. The command uses the following syntax:

`pridisplay number-of-octets-to-display line`

Syntax element	Description
`number-of-octets-to-display`	Specifies the number of octets in the PRI messages to display. Specify 0 (zero) to disable the display.
`line`	The PRI line to display. Specify 0 (zero) to display any line.

Example:

```
e1-1/15> pridisplay 128 0
Display the first 128 bytes of PRI messages
e1-1/15> PRI-XMIT-7: 10:37:00: 4 of 4 octets
800F1020: 00 01 01 73                                           ...s
PRI-RCV-7: 10:37:00: 4 of 4 octets
800F3CA0: 00 01 01 73                                           ...s
PRI-XMIT-7: 10:37:10: 4 of 4 octets
800F1020: 00 01 01 73                                           ...s
PRI-RCV-7: 10:37:10: 4 of 4 octets
800F3CA0: 00 01 01 73                                           ...s
PRI-XMIT-7: 10:37:20: 4 of 4 octets
800F1020: 00 01 01 73                                           ...s
PRI-RCV-7: 10:37:20: 4 of 4 octets
800F3CA0: 00 01 01 73                                           ...s
PRI-XMIT-7: 10:37:30: 4 of 4 octets
800F38E0: 00 01 01 73                                           ...s
PRI-RCV-7: 10:37:30: 4 of 4 octets
800F3CE0: 00 01 01 73                                           ...s
pridisplay 0
PRI message display terminated
```

RADacct

Description: Displays RADIUS accounting information. The RADacct command displays very few messages if RADIUS Accounting is functioning correctly.(`RADif` displays more detailed information for troubleshooting RADIUS-related issues.) The RADacct command is a toggle that alternately enables and disables the debug display.

Usage: Enter **radacct** at the command prompt.

Example:

```
admin> radacct
RADACCT debug display is ON
```

A user hangs up and a stop record is generated.

```
RADACCT-147:stopRadAcct
```

The following message indicates that there is some load on the network, and the sending of a stop record is delayed. This is not necessarily an indication of a problem.

```
RADACCT-147:_endRadAcct: STOP was delayed
```

RADif

Description: Displays RADIUS-related messages. RADif is a powerful diagnostic command, because it displays RADIUS messages the MAX TNT receives as well as messages that it sends. Output from `RADif`, in conjunction with running your RADIUS daemon in debug mode (using the `-x` option), gives you virtually all the information you need to clarify issues relating to user authentication.

You can also validate the IP port that you have configured (or think you have configured), and the user name that is being sent by the client.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **radif** at the command prompt.

Example: Following are messages you might see for a successful RADIUS authentication:

```
RADIF: authenticating <8:my_name> with PAP
RADIF: _radiusRequest: id 41, user name <9:my_name>
RADIF: _radiusRequest: challenge len = <0>
```

The IP address and RADIUS Daemon Authentication port are displayed:

```
RADIF: _radiusRequest: socket 5 len 89 ipaddr 01010101 port
65534->1645
RADIF: _radCallback
RADIF: _radCallback, buf = B05BBFA0
```

The response is sent back from RADIUS. In this case, the user `my_name` has passed authentication. Following is a list of the most common responses:

```
1 - Authentication Request
2 - Positive acknowledgement
3 - Rejection
4 - Accounting request
5 - Accounting response
7 - Password change request
8 - Password change positive acknowledgement
9 - Password change rejection
11 - Access challenge
29 - Password - next code
30 - Password New PIN
```

```
   31 - Password Terminate Session
   32 - Password Expired
RADIF: _radCallback, authcode = 2
RADIF: Authentication Ack
```

After, authenticating a user, the RADIUS daemon sends the attributes from the user profile to the MAX TNT. The MAX TNT creates the user's Connection profile from these attributes, and RADif displays them. (See the *MAX TNT RADIUS Configuration Guide* for a complete list of attribute numbers.)

```
RADIF: attribute 6, len 6, 00 00 00 02
RADIF: attribute 7, len 6, 00 00 00 01
RADIF: attribute 8, len 6, ff ff ff fe
RADIF: attribute 9, len 6, ff ff ff 00
RADIF: attribute 11, len 12, 73 74 64 2e
RADIF: attribute 12, len 6, 00 00 05 dc
RADIF: attribute 10, len 6, 00 00 00 00
RADIF: attribute 13, len 6, 00 00 00 01
RADIF: attribute 244, len 6, 00 00 11 94
RADIF: attribute 169, len 6, 00 00 11 94
RADIF: attribute 170, len 6, 00 00 00 02
RADIF: attribute 245, len 6, 00 00 00 00
RADIF: attribute 235, len 6, 00 00 00 01
```

A RADIUS Accounting Start packet is sent to the RADIUS Accounting Server (using port 1646):

```
RADIF: _radiusAcctRequest: id 42, user name <9:my_name>
RADIF: _radiusAcctRequest: socket 6 len 82 IP cf9e400b port 1646,
ID=42
RADIF: _radCallback
RADIF: _radCallback, buf = B05433C0
RADIF: _radProcAcctRsp: user:<9:my_name>, ID=42
```

RADservdump

Description: Use this command to verify the configuration you have set in the External-Auth profile.

Usage: Enter **`radservdump`** at the command prompt.

This does not display any information related to the configuration of either your RADIUS Authentication server or your RADIUS Accounting server.

Example: For the following example, the MAX TNT has been configured with two RADIUS servers, 1.1.1.1 and 2.2.2.2. The port has not been changed from its default of 1700.

```
admin> radservdump
Rad serv vars: port=1700,sockId=8
 0) clients=1010101
 1) clients=2020202
 2) clients=0
 3) clients=0
 4) clients=0
 5) clients=0
 6) clients=0
```

```
 7) clients=0
 8) clients=0
```

RADsessdump

Description: Displays the state of all RADIUS Accounting sessions.

Usage: Enter **radsessdump** at the command prompt.

Example:

```
admin> radsessdump
RadActSess:  state route sessID   nasPort authM  evTime
             loadd 00289 252365175 012032 local  523932
             loadd 00288 252365174 012032 local  523946
             loadd 00287 252365173 012032 local  523945
             loadd 00286 252365172 012032 local  523946
             loadd 00227 252355493 012032 local  370610
             loadd 00226 252355492 012032 local  370611
             loadd 00225 252355491 012032 local  370608
             loadd 00224 252355490 012032 local  370609
             loadd 00004 252332182 012032 none   29
             loadd 00003 252332181 012032 none   28
             loadd 00002 252332180 012032 none   27
             loadd 00001 252332179 012032 none   26
```

The RADsessdump command displays the following information:

Column Name:	Description
State	The state of the RADIUS accounting parameters and any accounting requests that have been sent. Values can be: • init—Initializing. No RADIUS accounting parameters have been loaded. • loadd—RADIUS accounting parameters have been loaded, but an accounting request either hasn't been issued or has failed. • start—All RADIUS accounting parameters are loaded. An accounting request has been issued. • done— Session over. No accounting request was issued, or the request failed. • stop—Session over. An accounting stop request has been issued.
Route	Internal route ID.
SessID	Session ID. This depends on the route ID.
NASPort	Statistics about the call. The first two digits indicate the type of call: 1 indicates a digital call, 2 indicates an analog call. The next two digits indicate the line on which the call was received. The last two digits indicate the channel on which the call was received.

Column Name:	Description
authM	Method of authentication.
evTime	Event time. This is a time stamp.

RADstats

Description: Displays a compilation of RADIUS Authentication and Accounting statistics.

Usage: Enter `radstats` at the command prompt.

Example:

```
admin> radstats
RADIUS authen stats:
```

In the following message, `A` denotes *Authentication*. `O` denotes *Other*. There were 612 Authentication requests sent and 612 Authentication responses received:

```
0  sent[A,O]=[612,15], rcv[A,O]=[612,8]
```

602 were authenticated successfully, and 18 were not:

```
timout[A,O]=[0,6], unexp=0, bad=18, authOK=602
```

In the next message, the IP address of the RADIUS server is 1.1.1.1, and the `curServerFlag` indicates whether or not this RADIUS server is the current authentication server. (You can have several configured RADIUS servers, but only one is current at any one time.) `0` indicates *no*. `1` indicates *yes*.

```
IpAddress 1.1.1.1, curServerFlag 1
RADIUS accounting stats:
```

The next message indicates that the MAX TNT sent 1557 Accounting packets and received 1555 responses (`ACKs` from the Accounting server). Therefore, the `unexp` value is `2`. This is not necessarily an indication of a problem, but might be the result of the MAX TNT timing out a particular session before receiving an `ACK` from the RADIUS server. Momentary traffic load might cause this condition. The value of `bad` is the number of packets that were formatted incorrectly by either the MAX TNT or the RADIUS server.

```
0  sent=1557, rcv=1555, timout=0, unexp=2, bad=0
```

In the next message, note that the Accounting server is different from the Authentication server. The Accounting and Authentication servers do not need to be running on the same host, although they can be.

```
IpAddress 2.2.2.2, curServerFlag 1
Local Rad Acct Stats:
```

The next two messages can be used to look for traffic congestion problems or badly formatted Accounting packets. Under typical conditions, you might see a few packets whose acknowledgments fail.

The following message indicates whether any RADIUS requests have been dropped by the MAX TNT. With this particular message, no requests were dropped. 1557 were sent successfully.

```
nSent[OK,fail]=[1557,0], nRcv=1557, nDrop[QFull,Other]=[0,0]
```

The following message indicates whether any session timeouts resulted from failure to receive RADIUS responses. The message also indicates responses that are received by the MAX TNT but do not match any expected responses. The MAX TNT keeps a list of sent requests, and expects a response for each request. In the following message, one response was received from the RADIUS server that did not match any of the requests that the MAX TNT had sent out. This might be caused by a corrupted response packet, or by the MAX TNT timing out the session before the response was received.

```
nRsp[TimOut,NoMatch]=[0,1], nBackoff[new,norsp]=[0,0]
```

The following messages display a summarized list of RADIUS server statistics.

```
Local Rad Serv Stats:
unkClient=0
index 0 #Sent = 0, #SendFail=0 badAuthRcv = 0, badPktRcv = 0
```

Reset

Description: This command resets the MAX TNT. When you reset the unit, it restarts and all active connections are terminated. All users are logged out and the default security level is reactivated. In addition, any active WAN lines are temporarily shut down due to loss of signaling or framing information. After a reset, the MAX TNT runs POST (power-on self-tests).

Usage: **`reset`**

Example: To reset the unit:

```
admin> reset
```

See Also: NVRAM

Revision

Description: Displays the serial number of the box.

Usage: Enter **`revision`** at the command prompt.

Example: In the following message, `7172461` is the serial number of the MAX TNT.

```
admin> revision
revision = 0 1 10 7172461
```

RoutMgr

Description: Displays information about the routing of incoming calls to either the Ethernet or modem ports. RoutMgr, when used in conjunction with Networki, can show valuable call routing information. If you have problems with users not connecting, and the incoming calls disconnect within one or two seconds of being presented to the MAX TNT, use RoutMgr and Networki to look for possible clues.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`routmgr`** at the command prompt.

Example:

```
admin> routmgr
ROUTMGR debug is now ON
ROUTMGR: buildIncomingRoute, port 0, phone <4990>
ROUTMGR: routMgrTask routeID=106, port=0, phone=4990
ROUTMGR-106: _matchPhoneNumber
```

There are no port limitations configured in the T1 profile:

```
ROUTMGR-106: _matchAnyPort
```

The next two messages show that the Bearer Capability in the ISDN setup message for the call indicates that it is a *voice* call, and that the call is routed to an available modem:

```
ROUTMGR-106: voice call
ROUTMGR: giving call to lan/hostif
```

At this point, the call is passed to other MAX TNT functions to continue the connection setup.

Following is output from RoutMgr when a call is cleared.

```
ROUTMGR: destroyRoute routeID = 106, cause = CLEAR
ROUTMGR-106: port is 59
ROUTMGR: deallocateCapabilityrouteID=106, capability=ALL
ROUTMGR: route 106 destroyed
```

SAR

Description: Shows packet bus statistics. Packet-bus traffic enters and exits a slot card (and shelf controller) by means of a chip called a SAR.

Usage: sar *-option*

Where ***-option*** is one of the following:

Option	Description
`-s`	Show SAR errors.
`-s -a`	Show all statistics.
`-s -i` *`shelf slot`*	Show SAR statistics for the indicated shelf and slot.
`-s -m`	Show SAR memory partition.
`-c`	Clear global statistics.
`-c -a`	Clear all statistics.
`-c -i` *`shelf slot`*	Clear statistics for indicated shelf and slot.
`-v` *`shelf slot`*	Display the SAR virtual circuit table for the specified shelf and slot.
`-l`	List open channels.
`-y`	Loop back cell statistics (shelf controller only).
`-z`	Send loopback cell (shelf controller only).

Example:

In the following example, an administrator checks for SAR errors and finds that there are none.

```
admin> sar -s

Ver  RxAlrt  RxStop RxRstrt NoRxBuf bsPrErr cmPrErr  busErr NoTxMBx NoRxMBx  s
8       0       0       0       0       0       0       0       0       0  6
```

In the next example, the administrator displays all the SAR statistics for the system:

```
admin> sar -s -a
SH/SL        Tx    TxDone   TxNoBuf        Rx     RxErr  RxUnFlow   RxOvRun    a
 1/ 1         8         8         0      5816         0         0         0    0
 1/11    736947    736947         0    736473         0         0         0    0
 1/16     27637     27637         0     27494         0         0         0    0
 1/17       160       160         0       160         0         0         0    0
 1/19    174588    174588         0    174588         0         0         0    0
 2/21    822669    822669         0    822653         4         0         0    0
 3/21   1109332   1109332         0   1109301         5         0         0    0
 4/21    145403    145403         0         0         0         0         0    0
 5/21    145403    145403         0         0         0         0         0    0
 6/21    145403    145403         0         0         0         0         0    0
 7/21    145403    145403         0         0         0         0         0    0
 8/21    145403    145403         0         0         0         0         0    0
 9/21    145403    145403         0         0         0         0         0    0
```

In the following example, the administrator displays SAR statistics for shelf 1, slot 17 (the master shelf-controller).

```
admin> sar -s -i 1 17
SH/SL        Tx    TxDone   TxNoBuf        Rx     RxErr  RxUnFlow   RxOvRun  a
1/17        160       160         0       160         0         0         0    0
```

SNTP

Description: Displays messages related to the Simple Network Time Protocol (SNTP) functionality of the MAX TNT. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`sntp`** at the command prompt.

Example: Following are three examples of messages displayed with SNTP enabled.

The MAX TNT accepts time from a configured NTP server. The following message appears if the MAX TNT does not accept a supplied time:

```
Reject:li= x stratum= y tx= z
```

The following message indicates that the MAX TNT accepts the time from a specified NTP server:

```
Server= 0 Time is b6dd82ed d94128e
```

Because the stored time is off by more than one second, it is adjusted:

```
SNTP: x Diff1= y Diff2= z
```

StackLimit

Description: If any MAX TNT function uses all but 128 or fewer of the bytes available for the stack, this command enables a checking routing that logs a warning to the Fatal-History log. The command is a toggle that alternately enables and disables the debug display.

Description: This command will enable a checking routine that will log a warning to the Fatal-History log whenever any MAX TNT function usage gets within 128 bytes from the end of the stack. The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`stacklimit`** at the command prompt.

TDM

Description: Used to set up or query the TDM bus.

Usage: **`tdm`** [**`-option`**] [**`itemA itemB`**] [**`connectionId`**]

where ***`-option`*** is one of the following:

Option	Description
`-a`	Allocate by first available. (Used when setting up a TDM connection to test).
`-c`	Connect channels.
`-d`	Disconnect a channel.
`-f`	Allocate a TDM channel by (f)ix-by-slot-item.
`-r`	Allocate a TDM channel by round robin.
`-s`	Display TDM manager statistics.
`-l`	List all connections.
`-t`	Toggle TDM manager debug output.
`-u`	Display TDM channel usage statistics.
`-?`	Display this summary.

The other syntax elements are:

Element	Description
`-x` *`number`*	Set the next TDM channel to check.
`itemA`	Logical address to connect from.
`itemB`	Logical address to connect to.
`connectionID`	ID of connection to disconnect.

Example: Following are some examples of output from the TDM command. (For more information about testing the TDM bus, see “Testing packet and TDM traffic” on page 6-4.)

```
admin> tdm -l
--id-- --cstate-- cnt  tdm#  ---src(A)---  ---dst(B)---
     1 connected    8    32  01:02:04/001  01:11:01/001
                         33  01:02:04/002  01:11:01/002
                         34  01:02:04/003  01:11:01/003
                         35  01:02:04/004  01:11:01/004
                         36  01:02:04/005  01:11:01/005
                         37  01:02:04/006  01:11:01/006
                         38  01:02:04/007  01:11:01/007
                         39  01:02:04/008  01:11:01/008
     2 connected   24    40  01:02:06/001  01:11:01/009
                         41  01:02:06/002  01:11:01/010
                         42  01:02:06/003  01:11:01/011
                         43  01:02:06/004  01:11:01/012
                         .
                         .
                         .

admin> tdm -s
        Number of total connections: 9
        Number of active connections: 9
        Number of available channels: 839
        Number of used channels: 185
        Number of disconnection errors: 0
        Number of bad received messages: 0
        Number of invalid events: 0
        Number of missing connections: 0
        Number of bad events: 0
        Number of bad states: 0

admin> tdm -u
(non-empty entries ONLY)
  timslot    nUsed  --currSrc--- --currDst---
       32        1  01:02:04/001 01:11:01/001
       33        1  01:02:04/001 01:11:01/001
       34        1  01:02:04/001 01:11:01/001
       35        1  01:02:04/001 01:11:01/001
       36        1  01:02:04/001 01:11:01/001
       37        1  01:02:04/001 01:11:01/001
       38        1  01:02:04/001 01:11:01/001
       39        1  01:02:04/001 01:11:01/001
       40        1  01:02:06/001 01:11:01/009
       .
       .
       .
```

TDMtst

Description: TDMtst runs on the HDLC card and tests the TDM bus. You can use it to verify communication between HDLC cards. Because the command tests byte-stream

communication on the TDM bus, which must use a known time slot, it requires some setup before it can verify TDM traffic. (for more information about testing the TDM bus, see "Testing packet and TDM traffic" on page 6-4.)

Usage: **`tdmtst`** ***-option***

where ***`-option`*** is one of the following:

Option	Description
`-o` *`channel physical-address logical-address`*	Open a TDM channel between the physical address and the logical address.
`-c channel`	Close the TDM channel.
`-e` *`channel count size`*	Send packets across the TDM bus on the open channel.
`-b` *`channel count size`*	Send packets across the TDM bus on the open channel.
`-x` *`channel string`*	Send the specified string over the TDM channel.
`-s`	Display the TDM test statistics.
`-t`	Toggle debug level.

TelnetDebug

Description: Displays messages as Telnet connections are attempted or established. The Telnet protocol negotiates several options as sessions are established, and TelnetDebug displays the Telnet option negotiations.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`telnetdebug`** at the command prompt.

Example: The following session shows a successful Telnet connection from the MAX TNT terminal server to another UNIX host.

```
admin> telnetdebug
TELNET debug is now ON
```

The far-end UNIX host has been contacted:

```
TELNET-4: TCP connect
```

For this Telnet session, the MAX TNT will support options 24 and 1. The UNIX host should respond with either `DO` or `WONT`:

```
TELNET-4: send WILL 24
TELNET-4: recv WILL 1
```

The UNIX host will support option 1:

```
TELNET-4: repl DO 1
```

The MAX TNT receives a request to support option 3:

```
TELNET-4: recv WILL 3
```

The MAX TNT will support option 3:

```
TELNET-4: repl DO 3
```

The UNIX host will support option 3:

```
TELNET-4: recv DO 3
```

The UNIX host will not support option 24:

```
TELNET-4: recv DONT 24
```

The MAX TNT will not support option 24:

```
TELNET-4: repl WONT 24
```

The UNIX host will support options 1 and 3:

```
TELNET-4: recv WILL 1
TELNET-4: recv WILL 3
```

TNTCall

Description: Places or clears a call. This command does not work for Frame Relay connections.

Usage: **`tntcall`** [**`-t`**] [**`-c`** ***`connection-profile`***] [**`-h`** ***`route-ID`***] [**`?`**]

Option	Description
`-t`	Toggle debug level.
`-c connection-profile`	Place a call with the specified Connection profile.
`-h route-ID`	Clear the channel associated with the specified route ID.
`-?`	Display this summary.

Example: The following output shows the MAX TNT answering an incoming call.

A call comes into the MAX TNT:

```
TNTCALL-649: call answer request
11675.96: TNTCALL-649: task got ANSWER event
```

The call is assigned to a device (in this case the HDLC channel in shelf 1, slot 16, Munich chip 4, channel 26), and is assigned a unique session ID:

```
TNTCALL-649: answer event, dest 1:16:04/26, channels 1  sess
318288918
```

The TDM connection is set up for the call:

```
TNTCALL: tdm 11918 state changed to 1
11675.98: TNTCALL-0: task got TDM STATE event
TNTCALL-649: TDM active, answering call
TNTCALL: allocated port 889 for dev 1:16:04/27
```

The call is successfully established:

```
TNTCALL-649: call complete, status 0, 1 channels
11677.39: TNTCALL-649: task got CALL COMPLETE event
TNTCALL-649: call sent to slot
```

After the session is disconnected or timed out, the MAX TNT begins tearing down the call:

```
TNTCALL-649: dead call
TNTCALL-649: dead call, destroy the route
TNTCALL-649: route destroyed
11732.58: TNTCALL-649: task got DESTROYED event
TNTCALL-649: cleaning up
```

TNTMP

Description: Displays information about MP and MP+ bundles and their channels. You can execute the TNTMP command on the shelf controller or on an HDLC card. You must first execute the Open command to open a session with the card.

`Usage: tntmp -i`

Example: To display information about MP and MP+ bundles and their channels:

```
admin> tntmp -i
mpBundleID=13 masterSlot=1/15 masterMpID=2 ifCount=2 rtIf=1/17:6
           routeID     slot    ifNum localIfNum  localMpID
                32     1/15        1          1          2
                33     9/ 2      193          1          2
```

This command works on HDLC cards as well. First, open a session with HDLC card, and then execute the TNTMP command. For example:

```
admin> open 1 15
hdlc-1/15> tntmp -i
mpBundleID=13 masterSlot=1/15 masterMpID=2 ifCount=2 rtIf=1/17:6
           routeID     slot    ifNum localIfNum  localMpID
                32     1/15        1          1          2
                33     9/ 2      193          1          2
```

In this example, the output shows a two-channel MP or MP+ bundle with the first channel in slot 1/15 and the second (slave) channel in slot 9/2. The command displays the following information:

Field	Description
`mpBundleID`	The globally known bundle ID for the whole system. If the connection adds channels for additional bandwidth on demand, the call for those channels is compared to the current bundle and assigned the same bundle ID as the other channels of the call.

Field	Description
masterSlot	The channel that was established as the base channel of the connection. After the MAX TNT authenticates a call that is not part of an existing bundle, it establishes the base channel of the connection. That channel becomes the *master* of the multilink connection.
masterMpID	The bundle ID at the master slot card. (The masterMpID is always the same as the localMpID for channels on the master slot card.)
ifCount	The number of channels in the bundle.
rtIf	The shelf/slot:id for the Route Logical Interface.
routeID	The globally known ID for each call.
slot	The shelf/slot numbers of the channels in the MP or MP+ bundle.
ifnum	Channel number on the master slot card.
localIfNum	The channel number on the local slot card. For HDLC cards, the channels are numbered 1–192. In the output in the example, the master slot (1/15) shows channel number 1. The interface number for the slave slot (9/2) is also 1, meaning the first channel on that card. However, at the master slot card, the slave interface number is mapped to a pseudo-interface number greater than 192, so it is not confused with channels on the master slot.
localMpID	The bundle ID known locally to the slot card.

TSshow

Description: Displays uptime and revision information about the MAX TNT. The Uptime command and the Software-Version parameter display the same information.

Usage: **tsshow** [**?**] [***uptime***] [***revision***]

Syntax element:	Description:
?	List all options.
uptime	Display system uptime.
revision	Display software and version currently running.

Example: Following are some samples of TSshow output:

```
admin> tsshow
Show what? Type 'tsshow ?' for help.
admin> tsshow ?
tsshow ?            Display help information
tsshow uptime       Display system uptime.
tsshow revision     Display system revision.
admin> tsshow uptime
system uptime: up 36 days, 9 hours, 59 minutes, 27 seconds
```

```
admin> tsshow revision
system revision: tntsr 2.0.0
```

TunnelDebug

Description: Displays messages related to setting up Generic Routing Encapsulation (GRE) tunnels on the MAX TNT. The command is a toggle that alternately enables and disables the debug display.You would normally use this command with the ATMPdebug command.

Usage: Enter **`tunneldebug`** at the command prompt.

Example: The following example shows an ATMP tunnel being set up:

```
TUNNELTNT.CB[1/7]: Event=Start-Tunnel SN=80
TUNNELTNT[1/7]: DUMP [Start-Tunnel] SN=80  MC=1/17/24/10052400
  HN=[] priHA=[200.67.1.254] secHA=[] Udp=5150 pass=[ascend]
  IP=141.111.40.55  Mask=255.255.255.255   IPX=00000000:000000000000
TUNNEL: createFAsession: priHA=[200.67.1.254]  secHA=[]  udpPort=5150
        ifNum=1/17/24/10052400  MajDev=7  password=ascend
        mcIpAddr=141.111.40.55/32
TUNNEL-411: Alloc 1019F660  Id=411 TN=411
TUNNEL-411: resolving 200.67.1.254, port=5150,  SN=411
TUNNEL-START: In progress
TUNNELTNT[1/7]: DUMP [Start-Tunnel-Rsp] SN=411  MC=1/7/4/10059440
  LocalSN=80  GlobalSN=411  Status=In progress
TUNNEL: _dnsCallback: name=[200.67.1.254],  ip=200.67.1.254  DNS=411
TUNNEL-411: tunnelSetStatus: status=Good completion
TUNNELTNT[1/7]: DUMP [Update-Tunnel] SN=411
  TunnelNumber=405  mcRtIf=1/7/4/10059440  HomeRtIf=0/0/0/0
  HomeAgent=200.67.1.254:5150  HomeNetwork=[]   Flags=10  AgentMode=2
  IP=141.111.40.55  Mask=255.255.255.255   IPX=00000000:000000000000
TUNNELTNT[1/7]: DUMP [Set-Status] SN=411  ErrorCode=0
TUNNELTNT[1/7]: DUMP [Start-Tunnel-Rsp] SN=411  MC=1/7/4/10059440
  LocalSN=80  GlobalSN=411  Status=In progress
```

TunnelSlot

Description: The command has little practical use other than as a tool for developmental engineering.

Update

Description: Modifies optional functionality of the MAX TNT. To enable some options, you must obtain a set of hash codes (supplied by an Ascend representative) that will enable the functionality in your MAX TNT. After each string is entered, the word *complete* appears, indicating that the MAX TNT accepted the hash code.

If you enter `update` without a text string modifier, the MAX TNT displays a list of current configuration information.

Usage: `update` [*`text_string`*]

Example:

```
admin> update
Host interfaces: 4
Net interfaces: 4
Port 1 channels: 255
Port 2 channels: 255
Port 3 channels: 255
Port 4 channels: 255
Field features 1: 182
Field features 2: 33
Field features 3: 54
Protocols: 1
```

```
admin> update 5 1023 12321312312312321
```

The following two messages indicate that the text strings were entered incorrectly:

```
update command: invalid arg 3!
update command: disallowed
```

The following message indicates that the MAX TNT accepted the update string:

```
update command: command complete.
```

WANdisplay

Description: Displays all packets received from, or sent to any of the WAN interfaces. Because WANdisplay output shows what the MAX TNT is receiving from and sending to the remote device, the information can be very helpful in resolving PPP negotiation problems.

If you enter the command while traffic through your MAX TNT is heavy, the resulting amount of output can make it tedious to find the information you're looking for. The screen might even display the message `----- data lost -----`, which just means that not all the output can be displayed on the screen.

Depending on the types of information you need to gather, you might prefer to use the WANdisplay command during a period of low throughput, or to use WANdsess, WANopen or WANnext to focus the display.

`Usage: wandisplay number-of-octets-to-display`

Enter **`wandisplay 0`** to disable the logging of this information.

Example: Following are several examples of WANdisplay output. Note that the bytes are displayed in hexadecimal format.

```
admin> wandisplay 24
Display the first 24 bytes of WAN messages
```

```
> RECV-272:: 1 octets @ 5E138F74
  [0000]: 0D
RECV-272:: 13 octets @ 5E13958C
  [0000]: 0A 41 63 63 65 70 74 3A 20 69 6D 61 67
XMIT-276:: 1011 octets @ 2E12D8A4
  [0000]: 7E 21 45 00 03 EE 54 2B 40 00 37 06 BA 09 CF 2B
  [0010]: 00 86 D0 93 91 90 1A 0A
admin> wandisplay 0
```

```
WAN message display terminated
```

WANdsess

Description: Shows WAN data as it is received and transmitted for a particular user. The WANdsess command is very similar to the WANdisplay command, but when you use WANdsess, the MAX TNT displays only incoming and outgoing packets for a specific user. WANdsess is particularly helpful on a MAX TNT with several simultaneous active connections. The command acts as a filter, allowing you to focus your troubleshooting.

Use the WANdsess command with host cards only. You must first execute the Open command to open a session with the modem or HDLC card.

Usage: **`wandsess session-name octets`**

Syntax element	Description
`session-name`	Name of a local Connection profile or a RADIUS user profile.
`octets`	Maximum number of octets to display per packet. If you specify 0 (zero), the MAX TNT does not display any data.

Example: To open a session with a modem card, and activate the display of WAN data for Tim's sessions:

```
admin> open 1 7
modem-1/7> wandsess tim
RECV-tim:300:: 1 octets @ 3E13403C
  [0000]: 7E 21 45 00 00 3E 15 00 00 00 20 7D 31 C2 D2
RECV-tim:300:: 15 octets @ 3E133A24
  [0000]: D0 7D B3 7D B1 B3 D0 7D B3 90 02 04 03 00 35
XMIT-tim:300:: 84 octets @ 3E12D28C
  [0000]: 7E 21 45 00 00 4E C4 63 00 00 1C 7D 31 17 5F D0
  [0010]: 93 90 02 D0 93 91 B3 00
```

Note that the bytes are displayed in hexadecimal format.

See Also: WANdisplay, WANopening

WanEventsStats

Description: Displays statistics about WAN events of interest on a host card.

Usage: First, open a session to a host card, then enter **`waneventstats`** at the command prompt.

Example:

```
modem-1/2> wanEventStats
Output:
_sendCachedData() Counts:
NullWanInfo 0
BufLen: 0
NullHandle: 0
BadState: 0
```

```
QueuingFails: 0
ToMbufFails: 0
SendOk: 0

_loseCachedData() Counts:
NoBuf: 0
LoseOk: 0

_cachePrioData() Counts:
BadData: 0
MallocFails: 0
PrevCache: 0
CacheOk: 0

WanInfo Instance Error Counts:
_wanBufferSent: 0
_wanBufferRcvd: 0
_wanBreakRcvd: 0
_modemEventHandlerInstanceMismatch: 0

WanInfo TxPending Error Counts: 0

wanSendData() Counts:
_wanSendDataOk: 1fd2e
_wanSendDataHighPriority: 1fd2e
_wanSendDataNormPriority: 0
_wanSendDataNoInpMbuf: 0
_wanSendDataBadLen: 0
_wanSendDataNormPrioNoBuf: 0
_wanSendDataNoRoute: 0
```

In this output, the following counters should always be set to zero (a non-zero value indicates an error condition):

```
NullWanInfo 0
BufLen: 0
NullHandle: 0
BadState: 0
NoBuf: 0
BadData: 0
_wanBufferSent: 0
_wanBufferRcvd: 0
_wanBreakRcvd: 0
_modemEventHandlerInstanceMismatch: 0
WanInfo TxPending Error Counts: 0
_wanSendDataNoInpMbuf: 0
_wanSendDataBadLen: 0
```

The rest of the counters can have non-zero values, although most of them indicate how busy the system is and should have small values. For example, the following counters record high-priority message caching events:

```
SendOk: 0
LoseOk: 0
CacheOk: 0
```

The next counters record send message requests. These are the only counters that record normal events rather than errors. The first _wanSendDataOk counter represents the count of all HDLC packets sent out, which may be quite a large number. The other two counters represent the two types of HDLC data, normal and high priority. Their sum should equal the value of _wanSendDataOk in the absence of errors. For example:

```
_wanSendDataOk: 1fd2e
_wanSendDataHighPriority: 1fd2e
_wanSendDataNormPriority: 0
```

The next counter records dropped normal priority messages. A non-zero value indicates the number of normal messages dropped due to lack of a buffer. To some extent this indicates how busy the system is, but because sessions have a buffer quota, it is possible to drop a normal message and increment this counter even when the system is not overloaded and when it is not out of buffers.

```
_wanSendDataNormPrioNoBuf: 0
```

The next counter reports requests to send a packet being processed after the session has been terminated. This is a normal occurrence when a call terminates during data transfer. (Its value should normally be relatively small but not necessarily non-zero.)

```
_wanSendDataNoRoute: 0
```

The following counters record the system's inability to obtain a DRAM or HDLC buffer for high priority message caching:

```
QueuingFails: 0
ToMbufFails: 0
MallocFails: 0
```

The following counter records high priority messages that have been dropped from the cache due to the arrival of another high priority message for the same session:

```
PrevCache: 0
```

WANopening

Description: Shows WAN data as it is received and transmitted during connection establishment for all users. The WANopening command is particularly helpful for troubleshooting connection problems in which users make the initial connection, but are disconnected within a few seconds. The output of WANopening is very similar to the output of WANdisplay, but WANopening only shows packets until the connection has been completely negotiated.

Use the WANopening command with host cards only. You must first execute the Open command to open a session with the modem or HDLC card.

Usage: `wanopening octets`

The *`octets`* value specifies the maximum number of octets to display per packet. If you specify 0 (zero), the MAX TNT does not log WAN data

Example: To open a session with a modem card, and activate the display of WAN data received and transmitted during connection establishment:

```
admin> open 1 7

modem-1/7> wanopening
Display the first 24 bytes of WAN messages
RECV-272:: 1 octets @ 5E138F74
  [0000]: 0D
RECV-272:: 13 octets @ 5E13958C
  [0000]: 0A 41 63 63 65 70 74 3A 20 69 6D 61 67

XMIT-276:: 1011 octets @ 2E12D8A4
  [0000]: 7E 21 45 00 03 EE 54 2B 40 00 37 06 BA 09 CF 2B
  [0010]: 00 86 D0 93 91 90 1A 0A
```

Note that the bytes are displayed in hexadecimal format.

See Also: WANdisplay, WANdsess

WANtoggle

Description: Displays messages from the WAN drivers on the MAX TNT, including the states of calls that are passed from the MAX TNT call routing routines as the connection is prepared to be passed to the Ethernet drivers.

If you enter the command while traffic through your MAX TNT is heavy, the resulting amount of output can make it tedious to find the information you're looking for. The screen might even display the message `----- data lost -----`, which just means that not all the output can be displayed on the screen. You might prefer to use this command during a period of low throughput.

The command is a toggle that alternately enables and disables the debug display.

Usage: Enter **`wantoggle`** at the command prompt.

Example: Following is a typical example of output produced by a modem call into the MAX TNT. After the incoming call is determined to be an analog call, a modem is directed to answer it.

```
WAN-389: wanOpenAnswer
WAN-389: modem redirected back to wan
WAN-389: Startup frame received
WAN-389: Detected unknown message
WAN-389: Detected ASYNC PPP message
WAN-389: wanRegisterData, I/F 58
```

The next two messages appear when the call is cleared.

```
WAN-389: wanCloseSession, I/F 58
```

The last message is not an indication of a problem. The modem clears the call a split second before the software releases its resources. The software does a check on the modem, which has already been released. This message is not an indication of a problem.

```
WAN-??: no modem assoc w WanInfo
```

Multishelf System Administration

6

This chapter covers the following topics:

Overview

The MAX TNT multiprocessor design handles encapsulation protocols as closely to the remote side as possible. T1 cards handle ISDN signaling and framing, HDLC cards handle synchronous PPP framing, and modem cards handle asynchronous PPP framing. Frame relay and ATMP sessions are all handled at the slot card that terminates the connection.

The shelf controller supports system-level commands that enable you to manage profiles, add and delete routes from the routing table, and monitor and manage the system as a whole. Each slot card supports commands that allow you to monitor and manage the individual card. With these card-level commands you can display T1 alarms and statistics on T1 cards, or display the IP route cache on host cards and display modem and HDLC statistics on the modem and HDLC cards.

To troubleshoot the MAX TNT, you need to use a combination of system-level and card-level commands.

Hardware overview

Data buses in the MAX TNT include a control bus, a TDM bus, and a packet bus.

The control bus

In a single-shelf system, the shelf controller communicates with slot cards over the control bus to perform such tasks as call setup, updating of the routing tables, updating or creating profiles, and bringing up and managing the cards.

The TDM bus

TDM is an isochronous protocol that multiplexes a byte stream by sending one byte every 125 µsec. The TDM bus supports up to 1024 TDM channels (time slots). Each time slot feeds one DS0 at 8 KHz. In the MAX TNT, time slots are numbered from 32 to 1023.

The packet bus

The packet bus is used for routing packets throughout the system. It is a 155 Mbps full-duplex, nonblocking, noncollision based path between the MAX TNT modules. Segment Assembly Reassembly (SAR) devices on each card are responsible for breaking packets into cells and reassembling the cells into packets on the receiving end.

Devices on the shelf controller, called cubits, switch the data on and off the packet bus. Each shelf controller contains one cubit for cards and one for the shelf controller.

How the MAX TNT answers calls

This section briefly describes how the MAX TNT routes a packet it receives on a network card, such as a T1 card, to its destination.

For this example, the call is a synchronous digital PPP call (from a Pipeline 75, for example) and contains TCP/IP packets destined for a host on the MAX TNT local Ethernet.

1 A synchronous digital PPP call comes in over an ISDN PRI.

2 The telephone company switch notifies the T1 card over the ISDN D-channel, of an impending call.

3 If appropriate, the MAX TNT CLID-or DNIS-authenticates the call.

4 The T1 card Munich chip processes the D-channel signaling information. It then uses the control bus to notify the shelf controller over the Control bus of the details of call, including its bandwidth and whether it is digital. The Munich chip also requests assignment of a TDM bus channel.

5 To route the call, the shelf controller refers to the call routing database to identify an appropriate slot card. It then:

- Allocates a TDM channel for the call.
- Informs the T1 and HDLC cards via the control bus to program the TDM Control Ram (CRAM) for the TDM time slot. (On a multishelf system, the master-shelf controller and the slave shelf-controller also program the Intershelf TDM Control Ram (ICRAM).)

6 Once the CRAM has been programmed, the T1 card forwards the HDLC byte stream, including HDLC header and HDLC payload (the IP packet), over the assigned TDM slot to the HDLC card without any processing.

7 When the HDLC card receives the HDLC frames, it strips off the HDLC headers and assembles a complete IP packet.

8 Once a complete IP packet is assembled, the HDLC card is ready to forward that packet over the packet bus to the appropriate slot card.

9 The HDLC card compares the destination IP address of the IP packet to destinations listed in its IP route cache. A cache hit is the result of a successful comparison. A cache miss is the result of an unsuccessful comparison.

- On a cache hit, the HDLC card is able to forward the IP packet directly over the packet bus to the appropriate slot card.
- On a cache miss, the HDLC card sends the IP packet to the shelf controller over the packet bus and the shelf controller determines where to route it. Once the shelf controller knows the target slot card and port, it routes the packet over the packet bus. It also broadcasts the new route (including the destination IP address, the target slot card, and the port) over the control bus to all the slot cards, so that they can update their route caches.

10 Once the HDLC card has the target destination slot card and port for the IP packet, the card's Segment Assembly and Reassembly (SAR) chip frames the IP packet in ATM cells for forwarding over the packet bus.

11 The SAR chip on the Ethernet card reassembles a complete IP packet from the cells received from the HDLC card. Once a complete IP packet is reassembled, it frames the IP packet for transmission on the Ethernet.

Multishelf system overview

The master shelf-controller uses as a control bus some of the bandwidth of the multishelf cable connecting the shelves. Only the master shelf-controller sends out control messages. When the slave shelf-controller receives those messages, it acts as proxy for the master shelf-controller and forwards each message to the proper slot card.

Multishelf systems use the same process for setting up calls as the one described in "How the MAX TNT answers calls" on page 6-2. However, if all of the modems on the modem card in the master system are in use, the master shelf-controller sends the control message across the cable to the slave shelf-controller, which forwards the message to its modem card.

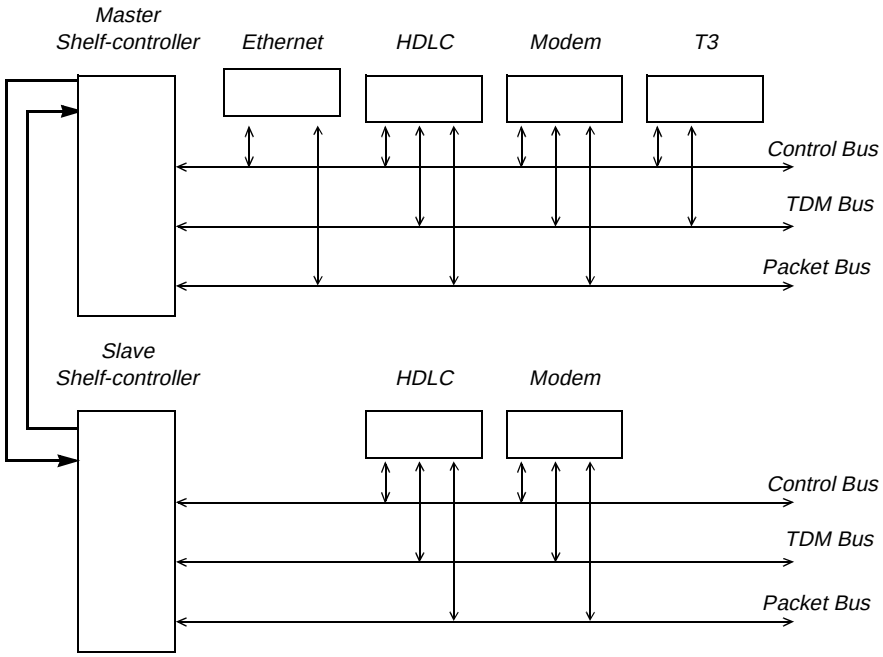

Figure 6-1. Multishelf system

The multishelf system has up to 1024 time slots globally, numbered 32 to 1023. It opens a time slot between the T3 channel in system 1 and a digital modem in system 2, and the session proceeds as it would on a single-shelf system.

Testing packet and TDM traffic

The MAX TNT provide two debug-level commands for testing packet and TDM traffic in a multishelf system. These commands are intended for troubleshooting purposes. Administrators do not usually need them unless there is some doubt that intershelf communications are occurring properly.

Testing packet bus traffic

The PBecho command tests the packet bus. You can use it to verify communication between the shelf controller and any slot cards that route packets, or to verify communication between two slot cards. The PBecho command operates similarly to Ping. It sends a packet to a known destination and echoes the packet back. The command uses the following syntax:

pbecho *shelf slot count size*

For example, if you are testing a two-shelf system in which the master-shelf rotary switch is set to 1 and the slave shelf is set to 2, the following command on the master shelf-controller sends a thousand 1500-byte packets to the slave shelf-controller:

```
admin> pbecho 2 17 1000 1500
pbus: Echo packets sent: 1000 rcvd: 1000 error: 0
```

(Slot 17 is the shelf-controller.) The output of the command indicates that the slave controller received 1000 packets and echoed them back to shelf 1. To further verify traffic across the

multishelf system, repeat the command a few times with different packet sizes. Then use the same command to send packets from the slave shelf-controller to the master.

Testing TDM traffic

On the TDM bus you can test multishelf communications between HDLC cards installed in different shelves. Normally, when the master shelf-controller is setting up a call, it selects an empty time slot on the TDM bus and writes the number of that time slot in the control RAM of both cards. When you use a debug command to verify byte-stream communication on the TDM bus, you must explicitly request that the master shelf-controller set up the time slot between HDLC channels.

To test TDM traffic, you must perform the following general steps:

1 Set up a TDM bus connection.

2 Open a TDM channel.

3 Test communications.

Setting up a TDM bus connection

When you set up a time slot between HDLC channels, you assign an address that has both physical (HDLC controller chip) and logical (HDLC channel number) address components. Each HDLC card has four controller chips (also called Munich chips), which are responsible for receiving HDLC frames and reassembling them into packets. Each controller chip controls 32 HDLC channels.

You must first find an available physical and logical HDLC channel to use for the test. Begin by using the TDM command to list the active TDM connections, as in the following example:

```
admin> tdm -l
--id-- --cstate-- cnt  tdm#  ---src(A)---  ---dst(B)---
     1 connected   24    32  01:01:01/001  01:16:01/001
                       33  01:01:01/002  01:16:01/002
                       34  01:01:01/003  01:16:01/003
                       35  01:01:01/004  01:16:01/004
                       36  01:01:01/005  01:16:01/005
                       37  01:01:01/006  01:16:01/006
                       38  01:01:01/007  01:16:01/007
                       39  01:01:01/008  01:16:01/008
                       40  01:01:01/009  01:16:01/009
```

The output shows that there is only one active TDM connection, between the T1 card in shelf 1, slot 1 and the HDLC card in shelf 1, slot 16. The connection is between line 1, channel 1 of the T1 card and line 1, channel 1 of the HDLC card. The system has assigned TDM channel-number 32 (representing a time slot) to the connection.

To test communication between shelves, you must allocate a physical and logical channel between two HDLC channels on separate shelves. In the following example, an administrator specifies a connection between the source HDLC channel on shelf 1, slot 16, HDLC chip 4, HDLC channel 32 and the destination HDLC channel on shelf 2, slot 16, HDLC chip 4, HDLC channel 32:

```
admin> tdm -c {{ 1 16 4 } 32} {{ 2 16 4 } 32}
```

This command tells the master shelf-controller to find an empty time slot and write the TDM channel number to control RAM in the source and destination HDLC cards.

To see which time slot has been allocated, use the `-l` option again. For example:

```
admin> tdm -l
--id-- --cstate-- cnt  tdm#  ---src(A)---  ---dst(B)---
     1 connected    1    64  01:16:04/032  02:16:04/032

     Total number of connections: 1
```

Opening a TDM channel

Once you have set up the TDM time slot between the two channels on separate shelves, you can test communications on the TDM bus by using the TDMtst command on each HDLC card. The following example demonstrates the procedure:

1 Open a session with the card installed in slot 16 of the master shelf:

```
admin> open 1 16
hdlc-1/16>
```

2 Open the allocated TDM time slot (64 in this example) and specify the physical and logical HDLC channel to use for the connection:

```
hdlc-1/16> tdmtst -o 64 4 32
```

3 Close the session with the card:

```
hdlc-1/16> quit
```

4 Open a session with the card installed in slot 16 of the second shelf:

```
admin> open 2 16
hdlc-2/16>
```

5 Open the allocated TDM time slot (64 in this example) and specify the physical and logical HDLC channel to use for the connection:

```
admin> open 2 16
hdlc-2/16> tdmtst -o 64 4 32
```

After you have established the connection, you can send data across it to test communications.

Testing communications

Once you have opened a TDM time slot between two unique addresses in the MAX TNT system, you can send a packet to the known destination and verify that it echoes the packet back. Proceed as in the following example:

1 Send packets across the TDM bus on the open time slot:

```
hdlc-2/16> tdmtst -e 64 100 1500
```

where 64 is the time slot, 100 is the number packets to send, and 1500 is the size of the packets to send.

You can execute the command from either HDLC card.

2 Once you have verified communications, you can close the time slot from the HDLC card session, or from the master shelf-controller interface.

To close the time slot from the HDLC card:

```
hdlc-2/16> tdmtst -c 64
```

Or, from the master shelf-controller:

```
admin> tdm -d 64
```

To make sure that the TDM traffic between the shelves is unimpeded, you should run the test a few times with different packet sizes and from other HDLC cards.

Creating User Profiles

7

This chapter covers the following topics:

Overview

User profiles are for MAX TNT system administration. Do not confuse them with Connection profiles. User profiles are used by administrators who need access to the MAX TNT command line interface to monitor or configure the unit. Connection profiles contain authentication and configuration information for a remote device or user and allow the remote user to connect to the MAX TNT for WAN or LAN access.

You can create any number of User profiles and fine-tune the privileges they allow. In addition to authentication and permission information, User profiles also contain parameters that affect how the user's environment appears at login.

The MAX TNT ships with two predefined User profiles, named Admin and Default. The Admin account is the super-user, with full read-write permissions. Default is set to the other extreme. It authorizes the minimal use of commands.

Many sites choose to create some administrative accounts in a read-only mode, to allow those users to check status windows, read log buffers, and execute diagnostic commands. You need at least one administrative account in read-write mode, but you may choose to create several such accounts.

Understanding the User profile parameters

Figure 7-1 describes common tasks you might have to perform to configure a User profile. The table includes a brief description of each task and lists the parameters you will use.

Table 7-1. Overview of User profile tasks

Task	Description of task	Associated parameters
Setting the name and password	When you create a new User profile with the New command, the system creates a default instance of the profile and reads it into the edit buffer. The name and password you assign to the profile represent a user or host name and a password used to authenticate that user at login.	Name Password
Activating the profile	The User profile is activated when you first create it. If you set Active-Enabled to No, the profile is not available for use.	Active-Enabled
Assigning permissions	Permissions control which actions the user who logs in with this profile can perform on the MAX TNT.	Allow-Termserv Allow-System Allow-Diagnostic Allow-Update Allow-Password Allow-Code
Logging the user out when idle	With the Idle-Timeout setting, you can specify the number of seconds a Telnet session can remain logged in with no keyboard activity.	Idle-Timeout
Setting the command-line prompt	The default command-line prompt is `TNT>`. If you set the prompt to an asterisk, the MAX TNT uses the name parameter as the prompt. For example, for the `admin` User profile, the prompt would be `admin>`.	Prompt
Specifying which status windows are displayed at login	You can display status windows by default at login, and you can specify what information should be displayed initially in the top, bottom, and left windows.	Default-Status Left-Status Top-Status Bottom-Status
Defining which log messages will be displayed	You can specify that log messages should be displayed immediately in the interface, instead of written to a log. You can also specify at which level the immediate display should begin. The lowest level is `none`, indicating that no messages should be displayed in the command-line interface. The highest level is `debug`.	Log-Message-Level

Understanding command permissions

Permissions control which actions the user who logs in with a particular profile can perform on the MAX TNT. Each permission enables the use of a command *class*. When you use the Help command to display available commands, the left column shows command names, and the right column shows the command class. For example:

```
admin> ?
?                               ( user )
arptable                        ( system )
auth                            ( user )
briChannels                     ( system )
cadslLines                      ( system )
callroute                       ( diagnostic )
cgCtrl                          ( system )
clear                           ( user )
clock-source                    ( diagnostic )
clr-history                     ( system )
connection                      ( system )
dadslLines                      ( system )
date                            ( update )
debug                           ( diagnostic )
delete                          ( update )
device                          ( diagnostic )
dir                             ( system )
dircode                         ( system )
dnstab                          ( system )
ds3AtmLines                     ( system )
ether-display                   ( diagnostic )
fatal-history                   ( system )
[More? <ret>=next entry, <sp>=next page, <^C>=abort]
```

Typically, read-write accounts enable the System command class. They might also enable the Update and Code command classes. Read-only accounts might be limited to the Diagnostic command class. Table 7-2 shows the commands associated with each permission:

Table 7-2. Permissions and associated commands

Permission	Command class	Commands in this class	
N/A (always enabled)	User	? Auth Clear	Help Quit Whoami

Table 7-2. Permissions and associated commands (continued)

Permission	Command class	Commands in this class	
Allow-System	System	ARPtable BRIChannels CADSLlines Clr-History Connection DADSLlines Dir Dircode DNStab Fatal-History Get HDLC IGMP IPcache IP-Pools IProute Line List Log Modem	Netstat New OSPF Power Quiesce Read Refresh Screen SDSLlines Set Show Status SWANlines T1channels UDS3lines Userstat VDSLchannels Version View
Allow-Diagnostic	Diagnostic	Callroute Clock-Source Debug Device DS3ATMlines Ether-Display If-Admin NSlookup	OAMLoop Open Ping Rlogin Slot Telnet Traceroute Uptime
Allow-Update	Update	Date Delete Load Nvram	Reset Save Write
Allow-Code	Code	Format	Fsck
Allow-Termserv	Termserv. This permission enables the user to invoke the Terminal-Server command and use the terminal-server interface.	Terminal-Server	

Table 7-2. Permissions and associated commands (continued)

Permission	Command class	Commands in this class
Allow-Password	N/A	The Allow-Password permission enables the user to view passwords. If set to No, the user sees a row of asterisks instead of the actual configured password. If the administrator that backs up system configurations does not have the Allow-Password permission set to Yes, passwords are not saved as part of the configuration.

Sample User profiles

If you have administrative privileges, you can create any number of User profiles that grant other administrators various degrees of access to the system.

When you create a new profile by specifying its index on the command line, the Default profile is used as the template. In the following is an example, an administrator creates a read-write administrative login named Marco, which has access to System, Diagnostic, and Update command classes:

```
admin> new user admin
USER/admin read
admin> set name = marco
admin> set password = my-password
admin> set allow-password = yes
admin> set allow-code = no
admin> write
USER/marco written
```

Following is an example of creating a User profile named Test, which is based on the Admin profile but restricts some permissions and has a different password:

```
admin> new user admin
USER/admin read
admin> set name = test
admin> set password = test-pw
admin> set allow-termserv = no
admin> set allow-update = no
admin> set allow-code = no
admin> write
USER/admin written
```

In the following example, an administrator creates a profile that enables the user to use the terminal-server commands but not to perform any other actions:

```
admin> new user
USER/default read
admin> set name = techpubs
admin> set password  = january
admin> set allow-termserv= yes
admin> set prompt = *
admin> set log-display-level = none
admin> write
USER/techpubs written
```

To log in by means of the new profile:

```
admin> auth techpubs
Password: january
```

Customizing the environment for a User profile

In addition to authentication and permission information, User profiles also contain parameters that affect how the user's environment appears at login. You can customize the following areas:

- The system prompt
- Whether the status window is displayed by default
- Information contained in the status window
- The level of log messages displayed

Setting the system prompt

The default command-line prompt is `TNT>`. You configure the prompt with the Prompt parameter. An asterisk in this setting causes the MAX TNT to substitute the value of the profile's name parameter upon successful login. For example, for the Admin profile, the prompt would be as follows:

```
admin>
```

Specifying status window information

The MAX TNT generates a continuous stream of statistics about its activities. You can specify in a User profile whether these statistics should always be displayed when a user logs in using that profile, what the areas of the window should display by default, and the size of the status windows. Opening the status window requires an 80-column by 24-row VT100 window.

The contents of the status window are determined by the following parameters in a User profile (show with their default values):

- Left-Status = Connection-List
- Top-Status = General-Info
- Bottom-Status = Log-Window

The size of the status window are determined by the following parameters in a User profile (shown with their default values)

- Screen-Length = 24
- Status-Length = 18

See the *MAX TNT Reference Guide* for details of using these parameters.

Figure 7-1 shows the default contents for each area of the status window:

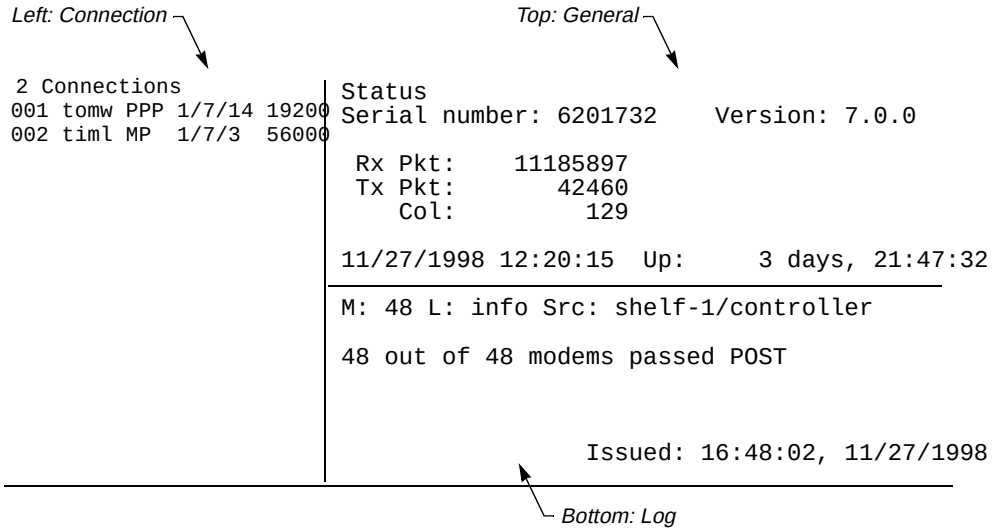

Figure 7-1. Information in the status window

Following is an example of configuring the User profile to display the status window upon login, and to show line information in the bottom area of the window. It also configures a larger terminal emulator window and status screens:

```
admin> read user test
USER/test read
admin> set default-status = yes
admin> set bottom-status = line-status
admin> set screen-length = 36
admin> set status-length = 30
admin> write
USER/test written
```

Note that Status-Length must be at least 6 lines smaller than Screen-Length.

Setting log levels for each login

You can configure the User profile to display a certain level of log messages immediately in the interface, in addition to writing them to a log file. Following is an example that causes critical, alert, and emergency messages to be displayed in the interface, interrupting whatever work might be going on at the prompt:

```
admin> read user test
USER/test read
admin> set log-display-level = critical
admin> write
USER/test written
```

Critical messages indicate that an interface has gone down or a security condition has been noted. Alert messages indicate that something undesirable has happened but probably will not prevent normal operation of the system. Emergency messages indicate that something undesirable has happened and will probably prevent normal operation.

Other levels include Error messages (an error condition has occurred), Warning messages (something out of the ordinary has occurred, such as a login failure), Notice (events in normal operation, such as a link going up or down), Info (changes that are not normally of interest), Debug (messages related to debugging configurations), and None (no messages are displayed).

Logging in as a different user

To login with a different User profile, use the Auth command, as in the following example:

```
admin> auth test
Password:@3wPZHd2
```

You must supply the password configured in the specified profile to be logged in as the user. Logging in as a different user can be helpful for verifying that the User profile permissions are correct.

Specifying a timeout for logins

You can specify a timeout period after which idle sessions on the MAX TNT disconnect. The default is 60 seconds. To configure an idle timeout, proceed as in the following example:

1 Read the System profile:

```
admin> read system
```

2 Specify an idle time period:

```
admin> set idle-logout=120
```

3 Write the profile:

```
admin> write
SYSTEM written
```

Finding the current user

To find out which User profile you are currently using, enter the Whoami command. For example:

```
admin> whoami
admin
```

SNMP Administration

8

This chapter covers these topics:

Overview

The MAX TNT supports SNMP (Simple Network Management Protocol) on a TCP/IP network. An SNMP management station that uses supported MIBs can query the MAX TNT, set some parameters, sound alarms when certain conditions appear in the MAX TNT, and so forth. The MAX TNT has its own SNMP password security (community strings), which you should set up to protect the MAX TNT from being reconfigured from an SNMP station.

The MAX TNT supports profiles that control which classes of events will generate traps to be sent to an SNMP manager, which SNMP managers may access the unit, and community strings to protect that access. This chapter describes SNMP support on the MAX TNT and also shows you how to set up the unit to work with SNMP.

SNMP support

This section describes the SNMP support on the MAX TNT.

Standard MIBS

This section describes the standard MIBs support on the MAX TNT.

RFC 1213 (MIB-II)

Enables you to monitor and configure basic components of the MAX TNT system, interfaces, and protocols. Note that the interface table in MIB-II is superseded by RFC 2233 (Interface MIB).

RFC 1253 (OSPF MIB)

Enables you to monitor and configure OSPF version 2.

RFC 1315 (Frame Relay MIB)

The Frame Relay MIB specifies SNMP MIB variables for Frame Relay DTEs. The MAX TNT HDLC cards support this MIB.

RFC 1317 (RS232 MIB)

Enables you to monitor and configure asynchronous or synchronous serial links with RS-232-like control signals.

RFC 1398 (Ethernet MIB)

Enables you to monitor the MAX TNT Ethernet interfaces.

RFC 1406 (DS1 MIB)

Enables you to query the state and configuration of T1 or E1 lines. The MAX TNT supports the all tables in this MIB except the `dsx1FracTable`.

RFC 1407 (DS3 MIB)

Enables you to query the state and configuration of T3 or E3 lines.

RFC 1695 (ATM MIB)

Enables you to manage the ATM interface on the MAX TNT ATM DS3 card. The MAX TNT supports the following groups in the ATM MIB related to network endpoints:

- (1) ATM Interface configuration group
- (2) ATM Interface DS3 PLCP group
- (3) ATM Interface TC Sublayer group
- (5) ATM Interface VCL configuration group
- (8) ATM Interface AAL5 VCC performance statistics group

Currently it is not possible to define new connections solely by using SNMP management, so many of the read-write and read-create parameters were changed to read-only.

RFC 1696 (Modem MIB)

The Modem MIB defines managed objects for modems. The MAX TNT supports all objects in the Modem MIB.

The Modem MIB defines a `mdmIndex` object whose value is used as an index into the tables defined in the MIB, with each modem in a managed system assigned a unique index value. This object is supported in the MAX TNT as a read-only Modem-Table-Index parameter in the Admin-State profile.

The value of this parameter is allocated by the system when it first detects the presence of a modem card.

The fact that the MAX TNT supports hot-swapable cards requires a relaxation of the MIB definition of the `mdmIndex` object in the same manner that RFC 1573 relaxes the `ifIndex` definition. The MIB definition of `mdmIndex` specifies that

- the index value must be in the range of 1 to mdmNumber, and
- the value must remain constant from one reinitialization of the network management agent to the next.

A modem card may be added to or removed from the MAX TNT without reinitializing the SNMP agent, which affects both of these definitions. For example, if a modem card is inserted into slot 1 of a new MAX TNT system, its 48 modems are allocated the index values 1 through 48. If another modem card is inserted into slot 3, its modems are allocated the index values 49 through 96. If the MAX TNT is rebooted, these values remain constant. If the modem card in slot 1 is removed and the MAX TNT is rebooted again, the index values for the modem card in slot 3 still remain constant with the range 49 through 96, even though the value of mdmNumber is now 48.

RFC 2233 (Interface MIB)

The MAX TNT supports the Interface MIB based on RFC 2233, which supersedes the SNMP MIB-II interface table defined in RFC1213. The interface table contains only the system's physical interfaces and nailed (permanent) interfaces.

The index value of an interface does not change following a system reset, and if an entry is removed from the interface table dynamically, its index value is not reused until the management station has been reinitialized. The interface table does not contain virtual circuit interfaces, such as a Frame Relay datalink configured on a channelized DS1 interface.

Table 8-1 describes MAX TNT support for RFC 2233.

Table 8-1. MAX TNT support for RFC 2233

RFC 2233 Table	Comment
`ifTable`	The ifTable from MIB-II (RFC 1213) and is fully supported on the MAX TNT.
`ifXTable`	The MAX TNT supports this table with the following exceptions: • The `OwnerString` object is not supported. • The `InterfaceIndexOrZero` object is not supported. • The 64-bit HighCounter objects are not supported. • The `ifPromiscuousMode` object is read-only.
`ifStackTable`	Not supported on the MAX TNT.
`ifRcvAddressTable`	Not supported on the MAX TNT.

Table 8-1. MAX TNT support for RFC 2233 (continued)

RFC 2233 Table	Comment
`ifTestTable`	Not supported on the MAX TNT.

Ascend enterprise MIBS

The Enterprise MIB is registered with the IANA (Internet Assigned Numbers Authority) as:

```
enterprises 529
```

with this value:

```
1.3.6.1.4.1.529
```

For the Ascend MIB hierarchy, see “Ascend MIB hierarchy” on page 8-16.

Ascend MIB (ascend.mib)

The Ascend MIB consists of the following groups:

- products (1)
- slots (2)
- hostTypes (3)
- advancedAgent (4)
- lanTypes (5)
- doGroup (6)
- hostStatus (7)
- console (8)
- systemStatusGroup (9)
- eventGroup (10)
- callStatusGroup (11)
- sessionStatusGroup (12)
- radiusGroup (13)
- mCastGroup (14)
- lanModemGroup (15)
- firewallGroup (16)
- wanDialoutPkt (17)
- powerSupply (18)
- multiShelf (19)
- miscGroup (20)
- asgGroup (21)
- flashGroup (22)
- configuration (23)
- atmpGroup (24)

- callLoggingGroup (25)
- srvcMgmtGroup (26)

Ascend ADSLCAP MIB (adslcap.mib)

The MAX TNT supports all objects in the ADSL MIB.

Ascend ADSL-CAP Profile MIB (mibcadslnet.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the ADSL-CAP profile in the command line interface.

Ascend ADSL-DMT Profile MIB (mibdadslnet.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the ADSL-DMT profile in the command line interface.

Ascend Advanced Agent MIB (advanced.mib)

The MAX TNT supports the Ascend Advanced MIB, previously called the WAN MIB. The Advanced MIB defines objects related to WAN lines, channels, and ports.

Ascend Answer Profile MIB (mibanswer.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the Answer-Defaults profile in the command line interface.

Ascend ATMP MIB (atmp.mib)

Enables you to configure and monitor Ascent Tunnel Management Protocol (ATMP) tunnels. For a complete description of ATMP, see RFC 2107, K. Hamzeh, “Ascend Tunnel Management Protocol - ATMP.”

Ascend Call MIB (call.mib)

Contains a table of entries for the status of each call in the system, including analog, digital, and Frame Relay-encapsulated calls. This MIB monitors the physical layer of the calls, including the slot and port. The Ascend Session MIB enables you to monitor the network layer of calls.

Ascend Call Logging MIB (call_log.mib)

Enables you to configure and monitor call logging, a RADIUS accounting-based feature for logging call information from the MAX TNT. Its main purpose is to duplicate accounting information for sites that want to keep accounting records separate from other groups that might need call-logging details to manage resources or troubleshoot call problems.

Ascend DS3 Profile MIB (mibds3net.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the T3 profile in the command line interface.

Ascend Event MIB (event.mib)

This read-only MIB enables you to monitor MAX TNT events. Includes connect progress and disconnect codes for calls

Ascend Firewall MIB (firewall.mib)

Enables you to dynamically configure Ascend Secure Access Firewalls that were created with Secure Access Manager (SAM). With this MIB, you can create or disable the firewall's dynamic rules.

Ascend Flash MIB (flash.mib)

Enables you to monitor the status of the MAX TNT flash cards, store or retrieve configuration files, or format the flash cards.

Ascend Frame Relay Profile MIB (mibfrmrl.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the Frame-Relay profile in the command line interface.

Ascend Internet Profile MIB (mibinet.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the Connection profile in the command line interface.

Ascend Lan Modem MIB (lmodem.mib)

Enables you to monitor the status of the MAX TNT digital modems, including the number available, number of bad or suspect modems, and usage statistics. It also allows you to disable individual modems.

Ascend Multicast MIB (mcast.mib)

This read-only MIB enables you to view the status of the multicast heartbeat monitor.

Ascend Multishelf MIB (ms.mib)

This MIB manages multishelf configuration, including whether the shelf is a master or a slave, its shelf number, and multishelf statistics.

Ascend Power Supply MIB (ps.mib)

This MIB manages the MAX TNT power supplies.

Ascend RADIUS MIB (radius.mib)

Enables you to view the status of Ascend RADIUS accounting and authentication servers, including client requests and the servers' responses. You can also use this MIB to mark a RADIUS server as the current server.

Ascend SDSL MIB (sdsl.mib)

The MAX TNT supports all objects in the SDSL MIB.

Ascend SDSL Profile MIB (mibsdslnet.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the SDSL profile in the command line interface.

Ascend Service Management MIB (srvcmgmt.mib)

Enables you to manage Dialed Number Information Service (DNIS) services on the MAX TNT. When the DNIS management mode is enabled, Network Management Stations (NMS) such as NavisAccess manage MAX TNT modem and HDLC resources.

Ascend Session MIB (session.mib)

Contains a table of entries for the status of each session in the system, including the IP address, type of session (PPP, MPP, Telnet, and so on), and MPP statistics.

Ascend UDS3 Profile MIB (mibuds3net.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the UDS3 profile in the command line interface.

Ascend VDSL Profile MIB (mibvdslnet.mib)

Part of the Ascend MIB Configuration group (group 23), this MIB corresponds to the VDSL profile in the command line interface.

Ascend WAN MIB (wan.mib)

Assigns the SNMP Object IDs (OIDs) for the WAN interfaces.

Ascend WAN Dialout MIB (wandialout.mib)

Enables you to monitor the packets the MAX TNT receives that causes it to dialout.

Ascend Enterpise traps

Defines Ascend-specific traps that alert NMS when certain events have occurred on the MAX TNT, such as when a Telnet session fails to authenticate, the MAX TNT is reset, or a Frame Relay DLCI is brought up or torn down.

Configuring SNMP access and security

The SNMP profile contains SNMP-readable information related to the unit itself and its SNMP security. There are two levels of security: community strings, which must be known by a community of SNMP managers to access the box, and address security, which excludes SNMP access unless it is initiated from a specified IP address.

These are the related parameters:

```
SNMP
   enabled = no
   read-community = public
   read-write-community = write
   enforce-address-security = no
   read-access-hosts = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 ]
   write-access-hosts = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 ]
   contact = ""
   location = ""
```

SNMP profile configuration overview

Table 8-2 provides some background information on tasks you may need to perform to configure SNMP on the MAX TNT. For complete details on each parameter, see the *MAX TNT Reference Guide*.

Table 8-2. SNMP profile configuration tasks

Task	Description
Enabling SNMP access	If the enabled parameter in the SNMP profile is set to No (the default), the MAX TNT cannot be accessed by SNMP utilities.
Setting community strings	The read-community parameter specifies the SNMP community name for read access (up to 32 characters), and the read-write-community parameter specifies SNMP community name for read/write access.
Setting up and enforcing address security	If the enforce-address-security parameter is set to No (its default value), any SNMP manager that presents the right community name will be allowed access. If it is set to Yes, the MAX TNT checks the source IP address of the SNMP manager and allows access only to those IP addresses listed in the read-access-host and write-access-host arrays. Each array can include up to five host addresses.
Specifying who to contact about problems and the location of the unit	The contact and location fields are SNMP readable and settable, and should indicate the person to contact about this unit, and its location.

Table 8-2. SNMP profile configuration tasks (continued)

Task	Description
Specifying a queue depth	The default queue depth for SNMP requests is zero, which means the packets will not be dropped, no matter how busy the SNMP subsystem gets. If the queue were to grow too large in an extremely loaded routing environment, the system could ultimately run out of memory. Valid values for the queue depth are 0–1024.

Sample SNMP profile

This example enables SNMP access, enforces address security, and prevents write access:

```
admin> read snmp
SNMP read

admin> list

enabled = no
read-community = public
read-write-community = write
enforce-address-security = no
read-access-hosts = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 ]
write-access-hosts = [ 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 ]
contact = ""
location = ""
admin> set enabled = yes
admin> set enforce-address-security = yes
admin> set read-access 1 = 10.2.3.4
admin> set read-access 2 = 10.2.56.123
admin> set queue-depth = 32
admin> write
SNMP written
```

Setting up SNMP traps

An SNMP trap is a mechanism for reporting system change in real time, such as reporting an incoming call. When a trap is generated by some condition, a traps-PDU (protocol data unit) is sent across the Ethernet to the SNMP manager.

You can configure the MAX TNT to send traps to an SNMP manager by specifying the address of the manager in a Trap profile. Traps can be enabled or disabled by class (error events, port state change events, or security events) or individually.

The following parameters relate to setting SNMP traps:

```
TRAP
   host-name* = ""
   community-name = ""
   host-address = 0.0.0.0
   alarm-enabled = yes
   security-enabled = no
```

```
port-enabled = no
slot-enabled=no
```

For details on the actual events that generate traps in the various classes, see the Ascend Enterprise MIB, or see the *MAX TNT Reference Guide*.

MAX TNT trap support

The MAX TNT does not support the `systemUseExceeded` trap.

Port-State change events are currently not applicable to the MAX TNT. These include:

- portInactive
- portDualDelay
- portWaitSerial
- portHaveSerial
- portRinging
- portCollectDigits
- portWaiting
- portConnected
- portCarrier
- portLoopback
- portAcrPending
- portDteNotReady
- portUseExceeded

In addition, TNT does not support billing features that include these traps:

- portUseExceeeded
- systemUseExceeded

Individual SNMP traps

Individual traps are enabled by default. The following parameters determine which traps are forwarded to an SNMP manager:

Parameter	Meaning
Slot-Enabled	The system generates a trap when a slot card is brought up or down.
Coldstart-Enabled	The system generates a trap when the MAX TNT reinitializes itself such that the configuration of the SNMP manager or the system itself might be altered.
Warmstart-Enabled	The system generates a trap when the MAX TNT reinitializes itself such that neither the configuration of SNMP manager or the system itself is altered.
Linkdown-Enabled	The system generates a trap when a failure occurs in a communication link between the unit and the SNMP manager.

Parameter	Meaning
Linkup-Enabled	The system generates a trap when the communication link between the unit and the SNMP manager comes back up.
Ascend-Enabled	(Also known as the Ascend Enterprise trap.) When both this parameter and Port-Enabled are set to Yes, a trap is generated to indicate a change of state in a host interface. All port connections are monitored in a state machine and reported via this trap.
Console-Enabled	The system generates a trap when the console has changed state. The console entry can be read to see what its current state is.
Use-Exceeded-Enabled	The system generates a trap when a specific port has exceeded the number of DS0 minutes allocated to it, or the system DS0 usage has been exceeded.
Password-Enabled	When both this parameter and Security-Enabled are set to Yes, all failed Telnet login attempts generate a trap.
FR-Linkup-Enabled	If both this parameter and Alarm-Enabled are set to Yes, a trap is sent whenever a DLCI is brought up.
FR-Linkdown-Enabled	If both this parameter and Alarm-Enabled are set to Yes, a trap is sent whenever a DLCI is brought down.
Event-Overwrite-Enabled	The system generates a trap when a new event has overwritten an unread event. This trap is sent only for systems which support Ascend's accounting MIB. Once sent, additional overwrites will not cause another trap to be sent until at least one table's worth of new events have occurred.
RADIUS-Change-Enabled	The system generates a trap when a new RADIUS server is being accessed. This trap returns the objectID and IP address of the new server.
Mcast-Monitor-Enabled	The system generates a trap when multicast heartbeat monitoring is configured and the system did not receive the configured number of heart-beat packets on a multicast interface.
LAN-Modem-Enabled	The system generates a trap when a digital modem is moved to the suspect list.
Dirdo-Enabled	The system generates a trap when a T-Online call comes in and no answer/subaddress has been received.
Slot-Profile-Change-Enabled	The system generates a trap when a Slot-State profile is created due to slot insertion, or the current-state transitions into Oper-State-Down, Oper-State-Up, Oper-State-Dump, or Oper-State-None states.
Power-Supply-Enabled	The system generates a trap when a power supply module is added or removed.
Multishelf-Enabled	The system generates a trap when a master shelf detects up/down state changes of the peer shelf in a multishelf configuration.
Authentication-Enabled	The system generates a trap when an authentication failure occurs.

SNMP trap configuration overview

Table 8-3 provides some background information on tasks you may need to perform to configure the MAX TNT to send SNMP traps. For complete details on each parameter, see the *MAX TNT Reference Guide*.

Table 8-3. SNMP trap configuration tasks

Task	Description	Associated parameters
Specifying the host running the SNMP manager	The Host-Name field is the index for the Trap profile, so it must contain a name. If DNS or YP/NIS is supported, it can contain the hostname of a system running an SNMP manager. If the host-address field contains an IP address, the specified name is not used to actually locate the host. The host-address can specify an IP address of the destination host. If DNS or YP/NIS is not supported, it must contain the host's address.	Host-Name
The community string for communicating with the SNMP manager	The community name field must contain the community name associated with the SNMP PDU.	Community-Name
Classes of traps to be sent to the specified host	The next three fields specify whether the MAX TNT traps alarm events, security events, and port events and sends a trap-PDU to the SNMP manager. For a description of the events that generate these traps, see the *MAX TNT Reference Guide*.	Alarm-Enabled Security-Enabled Port-Enabled
Individual traps to be sent to the specified host	In addition to enabling whole classes of traps, you can specify individual SNMP traps to forward to an SNMP manager. Individual traps are enabled by default.	Slot-Enabled Coldstart-Enabled Warmstart-Enabled Linkdown-Enabled Linkup-Enabled Ascend-Enabled Console-Enabled Use-Exceeded-Enabled Password-Enabled FR-Linkup-Enabled FR-Linkdown-Enabled Event-Overwrite-Enabled RADIUS-Change-Enabled Mcast-Monitor-Enabled LAN-Modem-Enabled Dirdo-Enabled Slot-Profile-Change-Enabled Power-Supply-Enabled Multishelf-Enabled Authentication-Enabled

Table 8-3. SNMP trap configuration tasks (continued)

Task	Description	Associated parameters
Enabling multishelf traps	Both the master shelf and slave shelf can forward SNMP traps if the multishelf link between them is down. If the link is down because the master shelf is powered down or reset, the slave shelf forwards a trap. If the link is down because the multishelf cables are disconnected, both the master and slave shelves can forward a trap.	Slot-Enabled

Example SNMP trap configuration

In the following example, the host-name is used only as a profile index, not to locate the actual host on the network. A community name is specified, security-class traps are added to the default alarm-class traps, and this host receives a trap if the multishelf link goes down.

```
admin> new trap
TRAP/"" read
admin> list
host-name* = ""
community-name = ""
host-address = 0.0.0.0
alarm-enabled = yes
security-enabled = no
port-enabled = no
slot-enabled = no
admin> set host-name = security-traps
admin> set community-name = Ascend
admin> set host-address = 10.2.3.4
admin> set security-enabled = yes
admin> set slot-enabled = yes
admin> write
TRAP/security-traps written
```

Because security traps and the Password-Enabled and Authentication-Enabled individual traps are enabled, two traps are sent when either of the related conditions occur. The individual trap provides additional information about the specific event that triggered the trap.

Multishelf traps

By default, the Multishelf MIB generates a trap when a multishelf link is down. If it is set to Disabled (2), the trap is not sent, regardless of Trap profile configurations. Slot-Enabled must be set to Yes in the Trap profile for the specified host to receive multishelf traps.

If traps are enabled on both the master and slave shelf controllers, a trap with the following OID might be generated to indicate multishelf link conditions:

.1.3.6.1.4.1.529.19.5.1.2.*X*

In this case, *X* in the OID is the number of the shelf that lost communication, and the trap value is 1 (idle).

A trap is reported by the master shelf-controller when the link is back up again. In this case, *X* in the OID is the destination shelf number, and the trap value is 4 (up). This trap is reported only by the master shelf to indicate that the entire multishelf system is up.

Managing SNMP interfaces

The MAX TNT supports the Interface MIB based on RFC 2233, which supersedes the SNMP MIB-II defined in RFC1213. The interface table contains only the system's physical interfaces and nailed (permanent) interfaces.

The index value of an interface does not change following a system reset, and if an entry is removed from the interface table dynamically, its index value is not reused until the management station has been reinitialized. The interface table does not It does not contain virtual circuit interfaces, such as a Frame Relay datalink configured on a channelized DS1 interface.

The If-Admin command is a diagnostic tool for managing SNMP interfaces. To see its usage:

```
admin> if-admin
usage: if-admin -a|d|l|r|u|? [ interface ]
       -a  list (a)available SNMP interface numbers
       -d  administratively (d)own an SNMP interface
       -l  (l)ist SNMP interface/device address mapping
       -r  (r)eset SNMP interface/device address mappings
       -u  administratively (u)p an SNMP interface
       -?  display this summary
```

To see a list of available SNMP interface numbers, use the `-a` option:

```
admin> if-admin -a
Available SNMP interface numbers
       118 - infinity
```

To see a list of all SNMP interface numbers assigned by the system:

```
admin>if-admin -l

SNMP-IF   DEVICE ADDRESS   STATUS

     1   -   { 1 17 1}     1
     2   -   { 1 1 1 }     1
     3   -   { 1 1 2 }     1
     4   -   { 1 1 3 }     1
     5   -   { 1 1 4 }     1
     6   -   { 1 1 5 }     1
     7   -   { 1 1 6 }     1
     8   -   { 1 1 7 }     1
     9   -   { 1 1 8 }     1
..
..
```

To bring an SNMP interface up or down, use the If-Admin command with the `-d` option, and specify the interface number. For example:

```
admin> if-admin -d 2
interface 2 state change forced
```

To bring a downed device back up, use the If-Admin command with the `-u` option, and specify the interface number. For example:

```
admin> if-admin -u 2
interface 2 state change forced
```

Initiating interface state changes

To bring an SNMP interface up or down, use the If-Admin command.

To bring an interface down:

```
admin> if-admin -d 2
interface 2 state change forced
```

To bring an interface up:

```
admin> if-admin -u 2
interface 2 state change forced
```

Resetting SNMP interface table sequentially

By default, the SNMP interface table is built as slot-cards are installed in the MAX TNT. The If-admin command `–r` option enables the administrator to reset the order of the table to be sequential based on slot number.

When you use the If-Admin command with the `–r` option, the order of the SNMP interface table is reset to a deterministic order. The T1 lines will appear in the SNMP interface table before the packet-passing interfaces such as Ethernet, modem, and HDLC cards. The T1 line interfaces will be ordered based on slot number order.

Note: You must reset the MAX TNT for the new order to take effect.

For example:

```
admin> if-admin -r
SNMP interface mappings reset.
Reset system in order to take effect.
```

Note: This command should not fail, but if for some reason it does, attempt it again. If it fails a second time, you should bring down all slot cards (`Slot –d`), remove all slot cards by using `Slot –r`, reset the system, and run the `If-admin –r` command again.

Ascend MIB hierarchy

Figure 8-1 illustrates the Ascend MIB hierarchy.

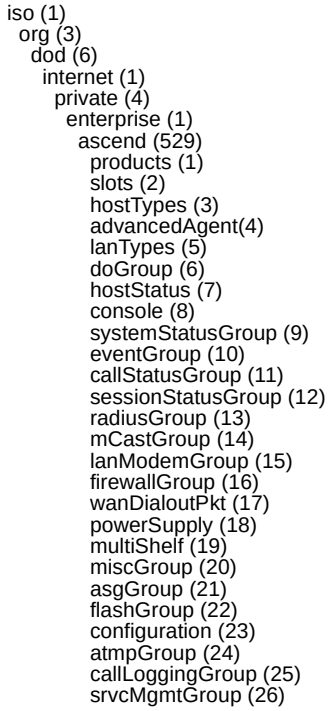

Figure 8-1. Ascend MIB hierarchy

products (1)

The `products` group is defined as:

`products ::= { enterprise ascend 1 }` with this value:
1.3.6.1.4.1.529.1

It contains the following objects:

multiband (1)
max (2)
pipeline (3)
max-tnt (4)
dslTnt (5)

slots (2)

The `slots` group is defined as:

`slots ::= { enterprise ascend 2 }` with this value:
1.3.6.1.4.1.529.2

It contains the following objects:

slotNumber(1)
slotTable(2)
  slotEntry (1)
    slotIndex (1)
    slotName (2)
    slotType (3)
    slotFixed (4)
    slotItems (5)
    slotSpecific (6)
    slotSerialNumber (7)
    slotStatus (8)
    slotLastChange (9)
slotItemTable (3)
  slotItemEntry (1)
    slotItemSlotIndex (1)
    slotItemIndex (2)
    slotItemFirstIf (3)
    slotItemIfCount (4)
    slotItemSpecific (5)
    slotItemStatus (6)
slotIfTable (4)
  slotifEntry (1)
    slotSlotIfIndex (1)
    slotIfSlotIndex (2)
    slotIfItemIndex (3)

hostTypes (3)

The `hostTypes` group is defined as:

`hostTypes ::= { enterprise ascend 3 }` with this value:
1.3.6.1.4.1.529.3

It contains the following objects:

hostTypeAny (1)
hostTypeDual (2)
hostTypeQuad (3)
hostTypeAim2 (4)
hostTypeAim6 (5)

advancedAgent (4)

The `advancedAgent` group is defined as:

`advancedAgent ::= { enterprise ascend 4 }` with this value:
1.3.6.1.4.1.529.1

It contains the following objects:

wanUseTrunkGroups(20)
wanLineTable (21)
  wanLineEntry (1)
    wanLineIfIndex (1)
    wanLineName (2)
    wanLineType (3)
    wanLineChannels (4)
    wanLineState (5)
    wanLineStateString (6)
    wanLineActiveChannels (7)
    wanLineUsage (8)
    wanLineHuntGrpPhoneNumber1 (9)
    wanLineHuntGrpPhoneNumber2 (10)
    wanLineHuntGrpPhoneNumber3 (11)
    wanLineAvailableChannels (12)
    wanLineSwitchedChannels (13)
    wanLineDisabledChannels (14)
    wanLineNailedChannels (15)
    wanLineOutOfServiceChannels (16)
wanLineChannelTable(22)
  wanLineChannelEntry (1)
    wanLineChannelIfIndex (1)
    wanLineChannelIndex (2)
    wanLineChannelState (3)
    wanLineChannelStateString (4)
    wanLineChannelErrorCount (5)
    wanLineChannelUsage (6)
    wanLineChannelTrunkGroup (7)
    wanLineChannelPhoneNumber (8)
    wanLineChannelSlot (9)
    wanLineChannelPort (10)
    wanLineChannelNailedState (11)
wanAvailableChannels (23)
wanSwitchedChannels (24)
wanDisabledChannels (25)
wanActiveChannels (26)
wanNailedChannels (27)
wanOutOfServiceChannels (28)

lanTypes (5)

The `lanTypes` group is defined as:

`products ::= { enterprise ascend 5}` with this value:
1.3.6.1.4.1.529.5

The Ascend MIB `lanTypes` group contains the following objects:

lanTypeAny (1)
lanTypeEthernet (2)
lanTypeEtherData (3)

doGroup (6)

The `doGroup` is defined as:

`products ::= { enterprise ascend 6 }` with this value:
1.3.6.1.4.1.529.6

The Ascend MIB `doGroup` contains the following objects:

```
doTable (1)
  doEntry (1)
    doSlotIndex (1)
    doItemIndex (2)
    doDial (3)
    doHangUp (4)
    doAnswer (5)
    doExtendBW (6)
    doContractBW (7)
    doBegEndRemoteLB (8)
doBegEndBERT (9)
doResynchronize (10)
```

hostStatus (7)

The `hostStatus` group is defined as:

`hostStatus ::= { enterprise ascend 7 }` with this value:
1.3.6.1.4.1.529.7

It contains the following objects:

```
hostStatusTable (1)
  hostStatusEntry (1)
    hostStatusSlotIndex (1)
    hostStatusItemIndex (2)
    hostStatusLocalName (3)
    hostStatusDialNum (4)
    hostStatusCallType (5)
    hostStatusCallMgm (6)
    hostStatusDataSvc (7)
    hostStatusCallState (8)
    hostStatusRemName (9)
    hostStatusChannels (10)
    hostStatusDuration (11)
```

console (8)

The `console` group is defined as:

`console ::= { enterprise ascend 8 }` with this value:
1.3.6.1.4.1.529.8

It contains the following objects:

```
consoleNumber (1)
consoleTable (2)
  consoleEntry (1)
    consoleIndex (1)
    consoleIf (2)
    consoleType (3)
    consoleSecurity (4)
    consoleSpecific (5)
```

systemStatusGroup (9)

The `systemStatusGroup` is defined as:

`systemStatusGroup ::= { enterprise ascend 9 }` with this value:
1.3.6.1.4.1.529.9

It contains the following objects:

sysAbsoluteStartupTime (1)
sysSecsSinceStartup (2)
sysMibVersionNum (3)
sysMibMinorRevNum (4)
sysConfigTftp (5)
  sysConfigTftpCmd (1)
  sysConfigTftpStatus (2)
  sysConfigTftpHostAddr (3)
  sysConfigTftpFilename (4)
  sysConfigTftpPort (5)
  sysConfigTftpParameter (6)
sysConfigRadius (6)
  sysConfigRadiusCmd (1)
  sysConfigRadiusStatus (2)
sysAbsoluteCurrentTime (7)
sysReset (8)
sysLoadName (9)
sysAuthPreference (10)
sysSPROM (11)
  sysSPROMSerialNumber (1)
  sysSPROMOptions1 (2)
  sysSPROMOptions2 (3)
  sysSPROMCountries1 (4)
resetStat (12)
  resetStatEther (1)
  resetStatWAN (2)
  resetStatAll (3)
sysLastRestartReason (13)

eventGroup (10)

The `eventGroup` is defined as:

`eventGroup ::= { enterprise ascend 10 }` with this value: 1.3.6.1.4.1.529.10

It contains the following objects:

```
eventMaximumNumberOfEvents (1)
eventOldestEventIdNumber (2)
eventLatestEventIdNumber (3)
eventTable(4)
  eventEntry (1)
    eventIdNumber (1)
    eventTimeStamp (2)
    eventType (3)
    eventCallReferenceNum (4)
    eventDataRate (5)
    eventSlotNumber (6)
    eventSlotLineNumber (7)
    eventSlotChannelNumber (8)
    eventModemSlotNumber (9)
    eventModemOnSlot (10)
    eventCurrentService (11)
    eventUserName (12)
    eventUserIPAddress (13)
    eventUserSubnetMask (14)
    eventDisconnectReason (15)
    eventConnectProgress (16)
    eventCallCharge (17)
    eventCalledPartyID (18)
    eventCallingPartyID (19)
    eventInOctets (20)
    eventOutOctets (21)
    eventMultiLinkID (22)
    eventXmitRate (23)
eventCurrentActiveCalls (5)
eventCurrentActiveSessions ( 6)
eventTotalCalls ( 7)
eventTotalSessions ( 8)
eventTotalCallsAnswered ( 9)
eventTotalCallsOriginated ( 10)
eventTotalCallsCleared ( 11)
eventTotalBaudRateChanges ( 12)
eventTotalServiceChanges ( 13)
eventTotalNameChanges ( 14)
eventTotalNoModems ( 15)
```

callStatusGroup (11)

The `callStatusGroup` is defined as:

`callStatusGroup ::= { enterprise ascend 11 }` with this value: 1.3.6.1.4.1.529.11

It contains the following objects:

callStatusMaximumEntries (1)
callStatusTable (2)
  callStatusEntry (1)
    callStatusIndex (1)
    callStatusValidFlag (2)
    callStatusStartingTimeStamp (3)
    callStatusCallReferenceNum (4)
    callStatusDataRate (5)
    callStatusSlotNumber (6)
    callStatusSlotLineNumber (7)
    callStatusSlotChannelNumber (8)
    callStatusModemSlotNumber (9)
    callStatusModemOnSlot (10)
    callStatusIfIndex (11)
    callSessionIndex (12)
    callStatusType (13)
    callStatusXmitRate (14)
    callStatusPortType (15)
callStatusHighWaterMark (3)
callCurrentAnalogOutgoing (4)
callCurrentAnalogIncoming (5)
callCurrentDigitalOutgoing (6)
callCurrentDigitalIncoming (7)
callCurrentFROutgoing (8)
callCurrentFRIncoming (9)
callTotalAnalogOutgoing (10)
callTotalAnalogIncoming (11)
callTotalDigitalOutgoing (12)
callTotalDigitalIncoming (13)
callTotalFROutgoing (14)
callTotalFRIncoming (15)
callActiveTable (16)
  callActiveEntry (1)
    callActiveCallReferenceNum (1)
    callActiveIndex (2)
    callActiveValidFlag (3)
    callActiveStartingTimeStamp (4)
    callActiveDataRate (5)
    callActiveSlotNumber (6)
    callActiveSlotLineNumber (7)
    callActiveSlotChannelNumber (8)
    callActiveModemSlotNumber (9)
    callActiveModemOnSlot (10)
    callActiveIfIndex (11)
    callActiveSessionIndex (12)
    callActiveType (13)
    callActiveXmitRate (14)
    callActivePortType (15)

sessionStatusGroup (12)

The `sessionStatusGroup` is defined as:

`sessionStatusGroup ::= { enterprise ascend 12 }` with this value: 1.3.6.1.4.1.529.12

It contains the following objects:

ssnStatusMaximumSessions (1)
sessionStatusTable (2)
  sessionStatusEntry (1)
    ssnStatusIndex (1)
    ssnStatusValidFlag (2)
    ssnStatusUserName (3)
    ssnStatusUserIPAddress (4)
    ssnStatusUserSubnetMask (5)
    ssnStatusCurrentService (6)
    ssnStatusCallReferenceNum (7)
sessionActiveTable (3)
  sessionActiveEntry (1)
    ssnActiveCallReferenceNum (1)
    ssnActiveIndex (2)
    ssnActiveValidFlag (3)
    ssnActiveUserName (4)
    ssnActiveUserIPAddress (5)
    ssnActiveUserSubnetMask (6)
    ssnActiveCurrentService (7)
mppActiveStatsTable (4)
  mppActiveStatsEntry (1)
    mppStatsMpID (1)
    mppStatsRemoteName (2)
    mppStatsQuality (3)
    mppStatsBandwidth (4)
    mppStatsTotalChannels (5)
    mppStatsCLU (6)
    mppStatsALU (7)
    mppStatsStartingTimeStamp (8)

radiusGroup (13)

The `radiusGroup` is defined as:

`radiusGroup ::= { enterprise ascend 13 }` with this value:
1.3.6.1.4.1.529.13

It contains the following objects:

radiusNumAuthServers (1)
radiusNumAcctServers (2)
radiusAuthStatsTable (3)
  radiusAuthStatsEntry (1)
    radAuthServerIndex (1)
    radAuthLoginRqstSent (2)
    radAuthOtherRqstSent (3)
    radAuthRqstTimedOut (4)
    radAuthOtherRqstTimedOut (5)
    radAuthRspRcvd (6)
    radAuthOtherRspRcvd (7)
    radAuthUnexpRspRcvd (8)
    radAuthBadRspRcvd (9)
    radAuthAckRspRcvd (10)
    radAuthHostIPAddress (11)
    radAuthCurrentServerFlag (12)
radiusAcctStatsTable (4)
  radiusAcctStatsEntry (1)
    radAcctServerIndex (1)
    radAcctRqstSent (2)
    radAcctRqstTimedOut (3)
    radAcctRspRcvd (4)
    radAcctUnexpRspRcvd (5)
    radAcctHostIPAddress (6)
    radAcctCurrentServerFlag (7)
radiusNewNASPortIDFormat (5)

mCastGroup (14)

The `mCastGroup` is defined as:

`mCastGroup ::= { enterprise ascend 14 }` with this value:
1.3.6.1.4.1.529.14

It contains the following objects:

heartBeatMulticastGroupAddress (1)
heartBeatSourceAddress (2)
heartBeatSlotTimeInterval (3)
heartBeatSlotCount (4)
heartBeatPacketCount (5)

lanModemGroup (15)

The `lanModemGroup` is defined as:

`lanModemGroup ::= { enterprise ascend 15 }` with this value:
1.3.6.1.4.1.529.15

It contains the following objects:

availLanModem (1)
availLanModemTable (2)
  availLanModemEntry (1)
    availLanModemSlotIndex (1)
    availLanModemPortIndex (2)
    availLanModemUsedCount (3)
    availLanModemBadCount (4)
    availLanModemLast32 (5)
suspectLanModem (3)
suspectLanModemTable (4)
  suspectLanModemEntry (1)
    suspectLanModemSlotIndex (1)
    suspectLanModemPortIndex (2)
    suspectLanModemUsedCount (3)
    suspectLanModemBadCount (4)
    suspectLanModemLast32 (5)
disabledLanModem (5)
disabledLanModemTable (6)
  disabledLanModemEntry (1)
    disabledLanModemSlotIndex (1)
    disabledLanModemPortIndex (2)
    disabledLanModemUsedCount (3)
    disabledLanModemBadCount (4)
    disabledLanModemLast32 (5)
deadLanModem (7)
deadLanModemTable (8)
  deadLanModemEntry (1)
    deadLanModemSlotIndex (1)
    deadLanModemPortIndex (2)
    deadLanModemState (3)
busyLanModem (9)
busyLanModemTable (10)
  busyLanModemEntry (1)
    busyLanModemSlotIndex (1)
    busyLanModemPortIndex (2)
    busyLanModemUsedCount (3)
    busyLanModemBadCount (4)
    busyLanModemLast32 (5)
    busyDirection (6)
suspectTrapState (11)

firewallGroup (16)

The `firewallGroup` is defined as:

`firewallGroup ::= { enterprise ascend 16 }` with this value:
1.3.6.1.4.1.529.16

It contains the following objects:

firewallStatus (1)
firewallControl (2)
  fwallCtrlRuleName (1)
  fwallCtrlExecute (2)
  fwallCtrlTimeOut (3)
  fwallCtrlExtAddr (4)
  fwallCtrlExtAddrMask (5)
  fwallCtrlExtPort (6)
  fwallCtrlExtPortMax (7)
  fwallCtrlIntAddr (8)
  fwallCtrlIntAddrMask (9)
  fwallCtrlIntPort (10)
  fwallCtrlIntPortMax (11)
  fwallCtrlRoutAddr (12)
  fwallCtrlAddrOpts (13)

wanDialoutPkt (17)

The `wanDialoutPkt` group is defined as:

`wanDialoutPkt ::= { enterprise ascend 17 }` with this value:
1.3.6.1.4.1.529.17

It contains the following objects:

```
wanDialoutPktTableSize (1)
wanDialoutPktMaxSize (2)
wanDialoutPktCount (3)
wanDialoutPktTable (4)
  wanDialoutPktEntry (1)
    wanDialoutPktIndex (1)
    wanDialoutPktTime (2)
    wanDialoutPktPhoneNumber (3)
    wanDialoutPktProtocolType (4)
    wanDialoutPktInfo (5)
```

powerSupply (18)

The `powerSupply` group is defined as:

`powerSupply ::= { enterprise ascend 18 }` with this value:
1.3.6.1.4.1.529.18

It contains the following objects:

```
powerSupplyCount (1)
powerSupplyTable (2)
  powerSupplyEntry (1)
    powerSupplyIndex (1)
    powerSupplyState (2)
    powerSupplyOperationalState (3)
powerSupplyStateTrapState (3)
powerSupplyOperationalStateTrapState (4)
```

multiShelf (19)

The `multiShelf` group is defined as:

`multiShelf ::= { enterprise ascend 19 }` with this value:
1.3.6.1.4.1.529.19

It contains the following objects:

```
myShelfNumber (1)
myShelfOperation (2)
masterShelfNumber (3)
multiShelfTableSize (4)
multiShelfTable (5)
  multiShelfTable (1)
    multiShelfIndex (1)
    multiShelfState (2)
    multiShelfResentFrames (3)
    multiShelfNLinkUp (4)
    multiShelfTxQs (5)
    multiShelfTxSeq (6)
    multiShelfRxSeq (7)
    multiShelfTimerValue (8)
multiShelfStateTrapState (6)
```

miscGroup (20)

The `miscGroup` is defined as:

`miscGroup ::= { enterprise ascend 20 }` with this value:
1.3.6.1.4.1.529.20

It contains the following objects:

```
miscGroupFRTable (1)
miscGroupFREntry (1)
   MiscGroupFRLMIIndex (1)
   MiscGroupFRLMIDlci (2)
```

flashGroup (22)

The `flashGroup` is defined as:

`flashGroup ::= { enterprise ascend 22 }` with this value:
1.3.6.1.4.1.529.22

It contains the following objects:

```
flashDevice (1)
  flashDevices (1)
  flashDeviceTable (2)
    flashDeviceEntry (1)
      flashDeviceIndex (1)
      flashDeviceController (2)
      flashDeviceSlot (3)
      flashDeviceSize (4)
      flashDeviceUsed (5)
      flashDeviceState (6)
      flashDeviceMaster (7)
      flashDeviceFormatStatus (8)
      flashDeviceDescription (9)
flashFileTable (2)
  flashFileEntry (1)
    flashFileIndex (1)
    flashFileController (2)
    flashFileCard (3)
    flashFileSize (4)
    flashFileStatus (5)
    flashFileName (6)
    flashFileChecksum (7)
    flashFileVersion (8)
    flashFileAccess (9)
    flashFileDateTimeStamp (10)
flashOperation (3)
    flashOperationStatus (1)
    flashOperationCommand (2)
    flashOperationHost (3)
    flashOperationDestFileName (4)
    flashOperationSrcFileName (5)
    flashOperationController (6)
    flashOperationCard (7)
    flashOperationLoadType (8)
```

configuration (23)

The `configuration` group is defined as:

`configuration ::= { enterprise ascend 23 }` with this value:
1.3.6.1.4.1.529.23

It contains the following objects:

```
mibinternetProfile (1)
mibframeRelayProfile (2)
mibanswerProfile (3)
mibud3NetworkProfile (4)
mibuds3NetworkProfile (5)
mibcadslNetworkProfile (6)
mibdadslNetworkProfile (7)
mibsdslNetworkProfile (8)
mibvdslNetworkProfile (9)
```

mibinternetProfile (1)

The `mibInternetProfile` has the value:
1.3.6.1.4.1.529.23.1

The `mibInternetProfile` in the `configuration` group contains the following objects:

MibinternetProfileTable (1)
MibinternetProfileEntry (1)
internetProfile__station (1)
internetProfile__active (2)
internetProfile__encapsulation_protocol(3)
internetProfile__called_number_type(4)
internetProfile__dial_number(5)
internetProfile__clid(6)
internetProfile__ip_options__ip_routing_enabled(7)
internetProfile__ip_options__vj_header_prediction(8)
internetProfile__ip_options__remote_address(9)
internetProfile__ip_options__local_address(10)
internetProfile__ip_options__routing_metric(11)
internetProfile__ip_options__preference(12)
internetProfile__ip_options__down_preference(13)
internetProfile__ip_options__private_route(14)
internetProfile__ip_options__multicast_allowed(15)
internetProfile__ip_options__address_pool(16)
internetProfile__ip_options__ip_direct(17)
internetProfile__ip_options__rip(18)
internetProfile__ip_options__route_filter(19)
internetProfile__ip_options__source_ip_check(20)
internetProfile__ip_options__ospf_options__active(21)
internetProfile__ip_options__ospf_options__area(22)
internetProfile__ip_options__ospf_options__area_type(23)
internetProfile__ip_options__ospf_options__hello_interval(24)
internetProfile__ip_options__ospf_options__dead_interval(25)
internetProfile__ip_options__ospf_options__priority(26)
internetProfile__ip_options__ospf_options__authen_type(27)
internetProfile__ip_options__ospf_options__auth_key(28)
internetProfile__ip_options__ospf_options__key_id(29)
internetProfile__ip_options__ospf_options__cost(30)
internetProfile__ip_options__ospf_options__down_cost(31)
internetProfile__ip_options__ospf_options__ase_type(32)
internetProfile__ip_options__ospf_options__ase_tag(33)
internetProfile__ip_options__ospf_options__transit_delay(34)
internetProfile__ip_options__ospf_options__retransmit_interval(35)
internetProfile__ip_options__ospf_options__non_multicast(36)
internetProfile__ip_options__multicast_rate_limit(37)
internetProfile__ip_options__multicast_group_leave_delay(38)
internetProfile__ip_options__client_dns_primary_addr(39)
internetProfile__ip_options__client_dns_secondary_addr(40)
internetProfile__ip_options__client_dns_addr_assign(41)
internetProfile__ip_options__client_default_gateway(42)
internetProfile__ip_options__tos_options__active(43)
internetProfile__ip_options__tos_options__precedence(44)
internetProfile__ip_options__tos_options__type_of_service(45)
internetProfile__ip_options__tos_options__apply_to(46)
internetProfile__ip_options__tos_filter(47)
internetProfile__ipx_options__ipx_routing_enabled(48)
internetProfile__ipx_options__peer_mode(49)
internetProfile__ipx_options__rip(50)
internetProfile__ipx_options__sap(51)
internetProfile__ipx_options__dial_query(52)
internetProfile__ipx_options__net_number(53)
internetProfile__ipx_options__net_alias(54)
internetProfile__ipx_options__sap_filter(55)
internetProfile__ipx_options__ipx_spoofing(56)
internetProfile__ipx_options__spoofing_timeout(57)
internetProfile__ipx_options__ipx_sap_hs_proxy(58)
internetProfile__ipx_options__ipx_header_compression(59)
internetProfile__bridging_options__bridging_group(60)
internetProfile__bridging_options__dial_on_broadcast(61)
internetProfile__bridging_options__ipx_spoofing(62)
internetProfile__bridging_options__spoofing_timeout(63)
internetProfile__bridging_options__bridge_type(64)
internetProfile__session_options__call_filter(65)
internetProfile__session_options__data_filter(66)
internetProfile__session_options__filter_persistence(67)
internetProfile__session_options__idle_timer(68)
internetProfile__session_options__ts_idle_mode(69)
internetProfile__session_options__ts_idle_timer(70)
internetProfile__session_options__backup(71)
internetProfile__session_options__secondary(72)
internetProfile__session_options__atmp_gateway(73)
internetProfile__session_options__max_call_duration(74)
internetProfile__session_options__vtp_gateway(75)
internetProfile__session_options__blockcountlimit(76)
internetProfile__session_options__blockduration(77)
internetProfile__session_options__max_atmp_tunnels(78)
internetProfile__session_options__max_vtp_tunnels(79)
internetProfile__session_options__redial_delay_limit(80)
internetProfile__session_options__ses_rate_type(81)
internetProfile__session_options__ses_rate_mode(82)
internetProfile__session_options__ses_adsl_cap_up_rate(83)
internetProfile__session_options__ses_adsl_cap_down_rate(84
internetProfile__session_options__ses_adsl_dmt_up_rate(85)
internetProfile__session_options__ses_adsl_dmt_down_rate(86
internetProfile__session_options__rx_data_rate_limit(87)
internetProfile__session_options__tx_data_rate_limit(88)
internetProfile__telco_options__answer_originate(89)
internetProfile__telco_options__callback(90)
internetProfile__telco_options__call_type(91)
internetProfile__telco_options__nailed_groups(92)
internetProfile__telco_options__ft1_caller(93)
internetProfile__telco_options__force_56kbps(94)
internetProfile__telco_options__data_service(95)
internetProfile__telco_options__call_by_call(96)
internetProfile__telco_options__billing_number(97)
internetProfile__telco_options__transit_number(98)
internetProfile__telco_options__expect_callback(99)
internetProfile__telco_options__dialout_allowed(100)
internetProfile__telco_options__delay_callback(101)
internetProfile__ppp_options__send_auth_mode(102)
internetProfile__ppp_options__send_password(103)
internetProfile__ppp_options__substitute_send_name(104)
internetProfile__ppp_options__recv_password(105)
internetProfile__ppp_options__link_compression(106)
internetProfile__ppp_options__mru(107)
internetProfile__ppp_options__lqm(108)
internetProfile__ppp_options__lqm_minimum_period(109)
internetProfile__ppp_options__lqm_maximum_period(110)
internetProfile__ppp_options__cbcp_enabled(111)
internetProfile__ppp_options__mode_callback_control(112)
internetProfile__ppp_options__delay_callback_control(113)
internetProfile__ppp_options__trunk_group_callback_control(11
internetProfile__ppp_options__split_code_dot_user_enabled(11
internetProfile__ppp_options__ppp_interface_type(116)
internetProfile__mp_options__base_channel_count(117)
internetProfile__mp_options__minimum_channels(118)
internetProfile__mp_options__maximum_channels(119)
internetProfile__mp_options__bacp_enable(120)
internetProfile__mpp_options__aux_send_password(121)
internetProfile__mpp_options__dynamic_algorithm(122)
internetProfile__mpp_options__bandwidth_monitor_direction(12
internetProfile__mpp_options__increment_channel_count(124)
internetProfile__mpp_options__decrement_channel_count(125)
internetProfile__mpp_options__seconds_history(126)
internetProfile__mpp_options__add_persistence(127)
internetProfile__mpp_options__sub_persistence(128)
internetProfile__mpp_options__target_utilization(129)

internetProfile__fr_options__frame_relay_profile(130)
internetProfile__fr_options__dlci(131)
internetProfile__fr_options__circuit_name(132)
internetProfile__fr_options__fr_direct_enabled(133)
internetProfile__fr_options__fr_direct_profile(134)
internetProfile__fr_options__fr_direct_dlci(135)
internetProfile__tcp_clear_options__detect_end_of_packet(136)
internetProfile__tcp_clear_options__end_of_packet_pattern(137)
internetProfile__tcp_clear_options__flush_length(138)
internetProfile__tcp_clear_options__flush_time(139)
internetProfile__ara_options__recv_password(140)
internetProfile__ara_options__maximum_connect_time(141)
internetProfile__comb_options__password_required(142)
internetProfile__comb_options__interval(143)
internetProfile__comb_options__base_channel_count(144)
internetProfile__comb_options__compression(145)
internetProfile__x25_options__x25_profile(146)
internetProfile__x25_options__lcn(147)
internetProfile__x25_options__x3_profile(148)
internetProfile__x25_options__max_calls(149)
internetProfile__x25_options__vc_timer_enable(150)
internetProfile__x25_options__x25EncapsType(151)
internetProfile__x25_options__auto_call_x121_address(152)
internetProfile__x25_options__reverse_charge(153)
internetProfile__x25_options__call_mode(154)
internetProfile__x25_options__answer(155)
internetProfile__x25_options__inactivity_timer(156)
internetProfile__x25_options__if_mtu(157)
internetProfile__x25_options__x25_rpoa(158)
internetProfile__x25_options__x25_cug_index(159)
internetProfile__x25_options__x25_nui(160)
internetProfile__x25_options__pad_banner(161)
internetProfile__x25_options__pad_prompt(162)
internetProfile__x25_options__pad_nui_prompt(163)
internetProfile__x25_options__pad_nui_pw_prompt(164)
internetProfile__x25_options__pad_alias1(165)
internetProfile__x25_options__pad_alias2(166)
internetProfile__x25_options__pad_alias3(167)
internetProfile__x25_options__pad_diag_disp(168)
internetProfile__x25_options__pad_default_listen(169)
internetProfile__x25_options__pad_default_pw(170)
internetProfile__eu_options__dce_addr(171)
internetProfile__eu_options__dte_addr(172)
internetProfile__eu_options__mru(173)
internetProfile__x75_options__k_frames_outstanding(174)
internetProfile__x75_options__n2_retransmissions(175)
internetProfile__x75_options__t1_retran_timer(176)
internetProfile__x75_options__frame_length(177)
internetProfile__appletalk_options__atalk_routing_enabled(178)
internetProfile__appletalk_options__atalk_static_ZoneName(179)
internetProfile__appletalk_options__atalk_static_NetStart(180)
internetProfile__appletalk_options__atalk_static_NetEnd(181)
internetProfile__appletalk_options__atalk_Peer_Mode(182)
internetProfile__usrRad_options__acct_type(183)
internetProfile__usrRad_options__acct_host(184)
internetProfile__usrRad_options__acct_port(185)
internetProfile__usrRad_options__acct_key(186)
internetProfile__usrRad_options__acct_timeout(187)
internetProfile__usrRad_options__acct_id_base(188)
internetProfile__calledNumber(189)
internetProfile__dhcp_options__reply_enabled(190)
internetProfile__dhcp_options__pool_number(191)
internetProfile__dhcp_options__maximum_leases(192)
internetProfile__sharedprof_options(193)
internetProfile__t3pos_options__x25_profile(194)
internetProfile__t3pos_options__max_calls(195)
internetProfile__t3pos_options__auto_call_x121_address(196)
internetProfile__t3pos_options__reverse_charge(197)
internetProfile__t3pos_options__answer(198)
internetProfile__t3pos_options__t3PosHostInitMode(199)
internetProfile__t3pos_options__t3PosDteInitMode(200)
internetProfile__t3pos_options__t3PosEnqHandling(201)
internetProfile__t3pos_options__t3PosMaxBlockSize(202)
internetProfile__t3pos_options__t3PosT1(203)
internetProfile__t3pos_options__t3PosT2(204)
internetProfile__t3pos_options__t3PosT3(205)
internetProfile__t3pos_options__t3PosT4(206)
internetProfile__t3pos_options__t3PosT5(207)
internetProfile__t3pos_options__t3PosT6(208)
internetProfile__t3pos_options__t3PosMethodOfHostNotif(209)
internetProfile__t3pos_options__t3PosPidSelection(210)
internetProfile__t3pos_options__t3PosAckSuppression(211)
internetProfile__t3pos_options__x25_rpoa(212)
internetProfile__t3pos_options__x25_cug_index(213)
internetProfile__t3pos_options__x25_nui(214)
internetProfile__t3pos_options__data_format(215)
internetProfile__t3pos_options__link_access_type(216)
internetProfile__framed_only(217)
internetProfile__altdial_number1(218)
internetProfile__altdial_number2(219)
internetProfile__altdial_number3(220)
internetProfile__x32_options__x32_profile(221)
internetProfile__x32_options__call_mode(222)
internetProfile__tunnel_options__profile_type(223)
internetProfile__tunnel_options__tunneling_protocol(224)
internetProfile__tunnel_options__max_tunnels(225)
internetProfile__tunnel_options__atmp_ha_rip(226)
internetProfile__tunnel_options__primary_tunnel_server(227)
internetProfile__tunnel_options__secondary_tunnel_server(228)
internetProfile__tunnel_options__udp_port(229)
internetProfile__tunnel_options__password(230)
internetProfile__tunnel_options__home_network_name(231)
internetProfile__tunnel_options__unused(232)
internetProfile__pri_numbering_plan_id(233)
internetProfile__vrouter(234)
internetProfile__atm_options__atm1483type(235)
internetProfile__atm_options__vpi(236)
internetProfile__atm_options__vci(237)
internetProfile__action__(238)
mibinternetProfile__tcp_clear_options__portTable (2)
internetProfile__tcp_clear_options__port__station (1)
internetProfile__tcp_clear_options__port__index__ (2)
internetProfile__tcp_clear_options__port (3)
mibinternetProfile__tcp_clear_options__hostTable (3)
internetProfile__tcp_clear_options__host__station (1)
internetProfile__tcp_clear_options__host__index__ (2)
internetProfile__tcp_clear_options__host (3)
mibinternetProfile__ipx_options__ipx_sap_hs_proxy_netTable (4)
internetProfile__ipx_options__ipx_sap_hs_proxy_net__station (1)
internetProfile__ipx_options__ipx_sap_hs_proxy_net__index__ (2)
internetProfile__ipx_options__ipx_sap_hs_proxy_net (3)

mibframeRelayProfile (2)

The `mibframeRelayProfile` has the value:
1.3.6.1.4.1.529.23.2

The `mibframeRelayProfile` in the `configuration` group contains the following objects:

mibframeRelayProfileTable (1)
mibframeRelayProfileEntry (1)
frameRelayProfile__fr_name (1)
frameRelayProfile__active (2)
frameRelayProfile__nailed_up_group (3)
frameRelayProfile__nailed_mode (4)
frameRelayProfile__called_number_type (5)
frameRelayProfile__switched_call_type (6)
frameRelayProfile__phone_number (7)
frameRelayProfile__billing_number (8)
frameRelayProfile__transit_number (9)
frameRelayProfile__link_mgmt (10)
frameRelayProfile__call_by_call_id (11)
frameRelayProfile__link_type (12)
frameRelayProfile__n391_val (13)
frameRelayProfile__n392_val (14)
frameRelayProfile__n393_val (15)
frameRelayProfile__t391_val (16)
frameRelayProfile__t392_val (17)
frameRelayProfile__MRU (18)
frameRelayProfile__dceN392_val (19)
frameRelayProfile__dceN393_val (20)
frameRelayProfile__link_mgmt_dlci (21)
frameRelayProfile__action (22)

mibanswerProfile (3)

The `mibAnswerProfile` has the value:
1.3.6.1.4.1.529.23.3

The `mibanswerProfile` in the `configuration` group contains the following objects:

mibanswerProfileTable (1)
mibanswerProfileEntry (1)
answerProfile__index (1)
answerProfile__use_answer_for_all_defaults (2)
answerProfile__force_56kbps (3)
answerProfile__profiles_required (4)
answerProfile__clid_auth_mode (5)
answerProfile__ppp_answer__enabled (6)
answerProfile__ppp_answer__receive_auth_mode (7)
answerProfile__ppp_answer__disconnect_on_auth_timeout (8)
answerProfile__ppp_answer__bridging_group (9)
answerProfile__ppp_answer__link_compression (10)
answerProfile__ppp_answer__mru (11)
answerProfile__ppp_answer__lqm (12)
answerProfile__ppp_answer__lqm_minimum_period (13)
answerProfile__ppp_answer__lqm_maximum_period (14)
answerProfile__mp_answer__enabled (15)
answerProfile__mp_answer__minimum_channels (16)
answerProfile__mp_answer__maximum_channels (17)
answerProfile__mp_answer__bacp_enable (18)
answerProfile__mpp_answer__enabled (19)
answerProfile__mpp_answer__dynamic_algorithm (20)
answerProfile__mpp_answer__bandwidth_monitor_direction (21)
answerProfile__mpp_answer__increment_channel_count (22)
answerProfile__mpp_answer__decrement_channel_count (23)
answerProfile__mpp_answer__seconds_history (24)
answerProfile__mpp_answer__add_persistence (25)
answerProfile__mpp_answer__sub_persistence (26)
answerProfile__mpp_answer__target_utilization (27)
answerProfile__fr_answer__enabled (28)
answerProfile__tcp_clear_answer__enabled (29)
answerProfile__ara_answer__enabled (30)
answerProfile__v120_answer__enabled (31)
answerProfile__v120_answer__frame_length (32)
answerProfile__x25_answer__enabled (33)
answerProfile__x25_answer__x25_profile (34)
answerProfile__x25_answer__x3_profile (35)
answerProfile__x25_answer__max_calls (36)
answerProfile__x25_answer__vc_timer_enable (37)
answerProfile__x25_answer__auto_call_x121_address (38)
answerProfile__x25_answer__reverse_charge (39)
answerProfile__x25_answer__x3_custom_prof (40)
answerProfile__comb_answer__enabled (41)
answerProfile__comb_answer__password_required (42)
answerProfile__comb_answer__interval (43)
answerProfile__comb_answer__compression (44)
answerProfile__eu_answer__euraw_enabled (45)
answerProfile__eu_answer__euui_enabled (46)
answerProfile__eu_answer__dce_addr (47)
answerProfile__eu_answer__dte_addr (48)
answerProfile__eu_answer__mru (49)
answerProfile__ip_answer__enabled (50)
answerProfile__ip_answer__vj_header_prediction (51)
answerProfile__ip_answer__assign_address (52)
answerProfile__ip_answer__routing_metric (53)
answerProfile__ipx_answer__enabled (54)
answerProfile__ipx_answer__peer_mode (55)
answerProfile__session_info__call_filter (56)
answerProfile__session_info__data_filter (57)
answerProfile__session_info__filter_persistence (58)
answerProfile__session_info__idle_timer (59)
answerProfile__session_info__ts_idle_mode (60)
answerProfile__session_info__ts_idle_timer (61)
answerProfile__session_info__max_call_duration (62)
answerProfile__x75_answer__enabled (63)
answerProfile__x75_answer__k_frames_outstanding (64)
answerProfile__x75_answer__n2_retransmissions (65)
answerProfile__x75_answer__t1_retran_timer (66)
answerProfile__x75_answer__frame_length (67)
answerProfile__framed_only (68)
answerProfile__action (69)

mibuds3NetworkProfile (5)

The `mibuds3Profile` has the value:
1.3.6.1.4.1.529.23.5

The `mibuds3NetworkProfile` in the `configuration` group contains the following objects:

mibuds3NetworkProfileTable (1)
 mibuds3NetworkProfileEntry (1)
  uds3NetworkProfile__shelf (1)
  uds3NetworkProfile__slot (2)
  uds3NetworkProfile__item (3)
  uds3NetworkProfile__name (4)
  uds3NetworkProfile__physical_address__shelf (5)
  uds3NetworkProfile__physical_address__slot (6)
  uds3NetworkProfile__physical_address__item_number (7)
  uds3NetworkProfile__enabled (8)
  uds3NetworkProfile__profile_number (9)
  uds3NetworkProfile__line_config__trunk_group (10)
  uds3NetworkProfile__line_config__nailed_group (11)
uds3NetworkProfile__line_config__route_port__slot_number__slot_number (12)
uds3NetworkProfile__line_config__route_port__slot_number__shelf_number (13)
uds3NetworkProfile__line_config__route_port__relative_port_number__relative_port_number (14)
  uds3NetworkProfile__line_config__activation (15)
  uds3NetworkProfile__line_config__call_route_info__shelf (16)
  uds3NetworkProfile__line_config__call_route_info__slot (17)
  uds3NetworkProfile__line_config__call_route_info__item_number (18)
  uds3NetworkProfile__line_config__line_type (19)
  uds3NetworkProfile__line_config__line_coding (20)
  uds3NetworkProfile__line_config__loopback (21)
  uds3NetworkProfile__action (22)

mibcadslNetworkProfile (6)

The `mibcadslNetworkProfile` has the value:
1.3.6.1.4.1.529.23.6

The `mibcadslNetworkProfile` in the `configuration` group contains the following objects:

mibcadslNetworkProfileTable (1)
  mibcadslNetworkProfileEntry (1)
    cadslNetworkProfile__shelf (1)
    cadslNetworkProfile__slot (2)
    cadslNetworkProfile__item (3)
    cadslNetworkProfile__name (4)
    cadslNetworkProfile__physical_address__shelf (5)
    cadslNetworkProfile__physical_address__slot (6)
    cadslNetworkProfile__physical_address__item_number (7)
    cadslNetworkProfile__enabled (8)
    cadslNetworkProfile__profile_number (9)
    cadslNetworkProfile__line_config__trunk_group (10)
    cadslNetworkProfile__line_config__nailed_group (11)
  cadslNetworkProfile__line_config__route_port__slot_number__slot_number (12)
  cadslNetworkProfile__line_config__route_port__slot_number__shelf_number (13)
  cadslNetworkProfile__line_config__route_port__relative_port_number__relative_port_number (14)
  cadslNetworkProfile__line_config__activation (15)
  cadslNetworkProfile__line_config__call_route_info__shelf (16)
  cadslNetworkProfile__line_config__call_route_info__slot (17)
  cadslNetworkProfile__line_config__call_route_info__item_number (18)
  cadslNetworkProfile__line_config__data_rate_mode (19)
  cadslNetworkProfile__line_config__max_up_stream_rate (20)
  cadslNetworkProfile__line_config__max_down_stream_rate (21)
  cadslNetworkProfile__action (22)

mibdadslNetworkProfile (7)

The `mibdadslNetworkProfile` has the value:
1.3.6.1.4.1.529.23.7

The `mibdadslNetworkProfile` in the `configuration` group contains the following objects:

mibdadslNetworkProfileTable (1)
  mibdadslNetworkProfileEntry (1)
    dadslNetworkProfile__shelf (1)
    dadslNetworkProfile__slot (2)
    dadslNetworkProfile__item (3)
    dadslNetworkProfile__name (4)
    dadslNetworkProfile__physical_address__shelf (5)
    dadslNetworkProfile__physical_address__slot (6)
    dadslNetworkProfile__physical_address__item_number (7)
    dadslNetworkProfile__enabled (8)
    dadslNetworkProfile__profile_number (9)
    dadslNetworkProfile__line_config__trunk_group (10)
    dadslNetworkProfile__line_config__nailed_group (11)
    dadslNetworkProfile__line_config__route_port__slot_number__slot_number (12)
    dadslNetworkProfile__line_config__route_port__slot_number__shelf_number (13)
    dadslNetworkProfile__line_config__route_port__relative_port_number__relative_port_number (14)
    dadslNetworkProfile__line_config__activation (15)
    dadslNetworkProfile__line_config__call_route_info__shelf (16)
    dadslNetworkProfile__line_config__call_route_info__slot (17)
    dadslNetworkProfile__line_config__call_route_info__item_number (18)
    dadslNetworkProfile__line_config__data_rate_mode (19)
    dadslNetworkProfile__line_config__max_up_stream_rate (20
    dadslNetworkProfile__line_config__max_down_stream_rate (21)
    dadslNetworkProfile__action (22)

mibsdslNetworkProfile (8)

The `mibsdslNetworkProfile` has the value:
1.3.6.1.4.1.529.23.8

The `mibsdslNetworkProfile` in the `configuration` group contains the following objects:

mibsdslNetworkProfileTable (1)
  mibsdslNetworkProfileEntry (1)
    sdslNetworkProfile__shelf (1)
    sdslNetworkProfile__slot (2)
    sdslNetworkProfile__item (3)
    sdslNetworkProfile__name (4)
    sdslNetworkProfile__physical_address__shelf (5)
    sdslNetworkProfile__physical_address__slot (6)
    sdslNetworkProfile__physical_address__item_number (7)
    sdslNetworkProfile__enabled (8)
    sdslNetworkProfile__profile_number (9)
    sdslNetworkProfile__line_config__trunk_group (10)
    sdslNetworkProfile__line_config__nailed_group (11)
    sdslNetworkProfile__line_config__route_port__slot_number__slot_number (12)
    sdslNetworkProfile__line_config__route_port__slot_number__shelf_number (13)
    sdslNetworkProfile__line_config__route_port__relative_port_number__relative_port_number (14)
    sdslNetworkProfile__line_config__activation (15)
    sdslNetworkProfile__line_config__call_route_info__shelf (16)
    sdslNetworkProfile__line_config__call_route_info__slot (17)
    sdslNetworkProfile__line_config__call_route_info__item_number (18)
    sdslNetworkProfile__line_config__max_rate (19)
    sdslNetworkProfile__line_config__unit_type (20)
    sdslNetworkProfile__action (21)

mibvdslNetworkProfile (9)

The `mibvdslNetworkProfile` has the value:
1.3.6.1.4.1.529.23.9

The `mibvdslNetworkProfile` in the `configuration` group contains the following objects:

```
mibvdslNetworkProfileTable (1)
  mibvdslNetworkProfileEntry (1)
    vdslNetworkProfile__shelf (1)
    vdslNetworkProfile__slot (2)
    vdslNetworkProfile__item (3)
    vdslNetworkProfile__name (4)
    vdslNetworkProfile__line_interface__enabled (5)
    vdslNetworkProfile__line_interface__answer_number_1 (6)
    vdslNetworkProfile__line_interface__answer_number_2 (7)
    vdslNetworkProfile__line_interface__max_rate (8)
    vdslNetworkProfile__line_interface__unit_type (9)
    vdslNetworkProfile__physical_address__shelf (10)
    vdslNetworkProfile__physical_address__slot (11)
    vdslNetworkProfile__physical_address__item_number (12)
    vdslNetworkProfile__action (13)
mibvdslNetworkProfile__line_interface__channel_configTable (2)
  mibvdslNetworkProfile__line_interface__channel_configEntry (1)
    vdslNetworkProfile__line_interface__channel_config__shelf (1)
    vdslNetworkProfile__line_interface__channel_config__slot (2)
    vdslNetworkProfile__line_interface__channel_config__item (3)
    vdslNetworkProfile__line_interface__channel_config__index (4)
    vdslNetworkProfile__line_interface__channel_config__nailed_group (5)
```

atmpGroup (24)

The `atmpGroup` group is defined as:

`atmpGroup ::= { enterprise ascend 24 }` with this value:
1.3.6.1.4.1.529.24

It contains the following objects:

atmpAgentMode (1)
atmpAgentType (2)
atmpAgentUDPPort (3)
atmpAgentGreMtu (4)
atmpAgentForceFragmentation (5)
atmpAgentHAIdleLimit (6)
atmpLastErrorGenerated (7)
atmpAgentSentErrorTo (8)
atmpLastErrorRecv (9)
atmpAgentRecvErrorFrom (10)
atmpEnableAtmpTraps (11)
atmpAgentNumberFATunnels (12)
atmpAgentNumberHATunnels (13)
atmpAgentNumberLocalTunnels (14)
atmpAgentTunnelHighWater (15)
atmpTunnelTable (16)
  atmpTunnelEntry (1)
    atmpTunnelIndex (1)
    atmpTunnelId (2)
    atmpHAIpAddress (3)
    atmpFAIpAddress (4)
    atmpTunneledProtocol (5)
    atmpTunnelType (6)
    atmpTunnelState (7)
    atmpMnIpAddress (8)
    atmpMnNetmask (9)
    atmpMnIpxNetAddress (10)
    atmpMnIpxNodeAddress (11)
    atmpHNProfileName (12)
    atmpHNMaxTunnels (13)
    atmpFAPrimaryHAAddress (14)
    atmpFASecondaryHAAddress (15)
    atmpFASsnStatusIndex (16)
    atmpFAUserName (17)
    atmpInPkts (18)
    atmpInOctets (19)
    atmpInErrPkts (20)
    atmpOutPkts (21)
    atmpOutOctets (22)
    atmpOutErrPkts (23)
    atmpPktsForcedToFragment (24)
    atmpPktsFailedFragment (25)

callLoggingGroup (25)

The `callLoggingGroup` is defined as:

`callLoggingGroup ::= { enterprise ascend 25 }` with this value: 1.3.6.1.4.1.529.25

It contains the following objects:

callLoggingNumServers (1)
callLoggingServerTable (2)
  callLoggingServerEntry (1)
    callLoggingServerIndex (1)
    callLoggingCurrentServerFlag (2)
    callLoggingServerIPAddress (3)
callLoggingStatus (3)
callLoggingPortNumber (4)
callLoggingKey (5)
callLoggingTimeout (6)
callLoggingIdBase (7)
callLoggingResetTime (8)
callLoggingStopPacketsOnly (9)
callLoggingRetryLimit (10)
callLoggingAssStatus (11)
callLoggingDroppedPacketCount (12)
callLoggingRadCompatMode (13)

srvcMgmtGroup (26)

The `srvcMgmtGroup` is defined as:

`callLoggingGroup ::= { enterprise ascend 26 }` with this value:
1.3.6.1.4.1.529.26

It contains the following objects:

```
dnisMgmt (1)
  dnisMgmtGlobalEnabled (1)
  dnisMgmtGlobalNumEntries (2)
  dnisMgmtGlobalLastChange (3)
  dnisMgmtGlobalTable (4)
    dnisMgmtGlobalEntry(1)
      dnisGlobalIndex (1)
      dnisGlobalPhoneNumber (2)
      dnisGlobalStatus (3)
      dnisGlobalCallsAccepted (4)
      dnisGlobalCallsDropped (5)
      dnisGlobalAction (6)
```

Using Administrative Profiles

9

This chapter covers the following topics:

Overview

The MAX TNT provides a number of profiles that either monitor administration information or enable the administrator to change the state of a slot, line, or device. (For discussion of profiles not directly related to system administration, for example, profiles related to configuring lines, connections, or calls, see the *MAX TNT Network Configuration Guide* or the *MAX TNT Hardware Installation Guide*.)

Following are the MAX TNT administrative profiles:

Profile	**Description**
Admin-State-Perm-If	SNMP Permanent Interface Admin State
Admin-State-Phys-If	SNMP Physical Interface Admin State

Profile	Description
ADSL-CAP-Stat	ADSL-CAP line status
ADSL-DMT-Stat	ADSL-DMT line status
Base	System version and enabled features
Call-Info	Active call information
Call-Logging	RADIUS-accounting based feature for logging call information from the MAX TNT
Device-State	Device Operational State
DS3-ATM-Stat	DS3 ATM status
Error	Fatal Error Log
FRDLCI-Stat	Frame Relay DLCI-PVC state
FRPVC-Stat	Frame Relay PVC state
IDSL-Stat	ISDL interface status
LAN-Modem	LAN modem disable state
Log	System event logging configuration
RADIUS-Stat	RADIUS status
SDSL-Status	SDSL interface status
Slot-Info	Slot information
Slot-State	Slot Operational State
Slot-Type	Slot Type profile
SNMP	SNMP profiles
SWAN-Stat	Serial WAN interface status
System	System-level parameters
T1-Stat	T1 and E1 line status
T3-Stat	T3 line status
Timedate	Current system time and date
Trap	SNMP trap destinations
UDS3-Stat	Unchannelized DS3 status
User	Administrative user accounts

For information about the parameters contained within each of these profiles, see the *MAX TNT Reference Guide*.

An administrative profile uses the same set of commands as does any configuration profile in the MAX TNT. For example:

```
admin> read t1-stat { 1 5 1}
T1-STAT/{ shelf-1 slot-5 1 } read
```

```
admin> list
physical-address* = { shelf-1 slot-5 1 }
line-state = active
channel-state = [ nailed-up nailed-up nailed-up nailed-up
nailed-up nailed-up n+
error-count=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ]
loss-of-carrier = False
loss-of-sync = False
ais-receive = False
yellow-receive = False
ber-receive = False
carrier-established = True
network-loopback = False
```

How the MAX TNT creates administrative profiles

The MAX TNT allocates SNMP interfaces when a card comes up for the first time. For example, the initial installation of a T1 card creates eight SNMP interfaces, one for each T1 line. Admin-State profiles are stored in NVRAM to keep state information over system resets, so a physical device keeps the same SNMP interface number across system reset or power failures.

Each physical interface in the system has an associated Admin-State-Phys-If profile and each nailed connection, such as a Frame Relay connection or a nailed PPP connection, has an associated Admin-State-Perm-If profile. These profiles store the object's desired state and SNMP interface number.

At system startup, the MAX TNT reads the Admin-State profiles. If the addressed device is not present in the system and has been replaced by a device of another type, the MAX TNT deletes that profile and creates a new one, with a new SNMP interface number. The next time the system is reset or power cycles, the old device's SNMP interface number becomes available for reassignment. This means that pulling a slot card does not free up interface numbers. When you reinstall the slot card, the same interface number is assigned. Also, pulling a slot card and replacing it with a slot card of another type does not free up the old interface numbers until the next power cycle or system reset.

For example, each T1 line has an Admin-State-Phys-If profile, and each of the 48 modems on a modem card has a profile. To read the Admin-State-Phys-If profile for the first T1 line in Slot 2, use the Read and List commands, as in the following example:

```
admin>read admin-state-phys-if {1 2 1}
ADMIN-STATE-PHYS-IF/{ shelf-1 slot-2 1 } read

admin>list

[in ADMIN-STATE-PHYS-IF/{ shelf-1 slot-2 1 }]
device-address* = { shelf-1 slot-2 1 }
slot-type = 8t1-card
snmp-interface = 34
modem-table-index = 0
desired-state = admin-state-up
desired-trap-state = trap-state-enabled
```

Using the Admin-State-Perm-If profile

The Admin-State-Perm-If profile holds information about the MAX TNT nailed interfaces. The system creates a profile for an active nailed interface and assigns it an interface index. For example:

```
admin> dir admin-state-perm
    21  08/28/1998 13:21:37  frswan1
    21  08/28/1998 13:21:37  frswan6
    27  08/28/1998 13:22:11  radius-frt1.1
    30  09/02/1998 15:38:07  tnt-e1-ds3a-uds3
    30  09/02/1998 17:31:42  tnt-e1-ds3a-ds3a
```

The Admin-State-Perm-If profile contains the following parameters (shown here with sample values):

```
[in ADMIN-STATE-PERM-IF/frswan1]
station* = frswan1
snmp-interface = 19
desired-state = admin-state-up
desired-trap-state = trap-state-enabled
inet-profile-type = 1
```

Parameter	Specifies
Station	Name of a nailed profile (PPP or Frame Relay), which may be a local Connection profile or a RADIUS profile.
SNMP-Interface	Interface table index assigned to the nailed interface whose state is stored in this profile.The system assigns a numeric value.
Desired-State	Desired administrative state of the addressed device. The system sets it to Admin-State-Down if an operator downs the device, or to Admin-State-Up if an operator attempts to bring up the device in normal operations mode. An operator can change the admin state by using SNMP SET commands, or the Slot or If-Admin commands.
Desired-Trap-Sate	Desired link up/down enable state of the interface.The system sets it to Trap-State-Enabled if an operator specifies that linkUp/linkDown traps should be generated for the interface, or to Trap-State-Disabled if an operator specifies that linkUp/linkDown traps should not be generated for the interface.
Inet-Profile-Type	If the nailed profile is a local profile (0) or a RADIUS profile (1).

Using the Admin-State-Phys-If profile

The Admin-State-Phys-If profile holds information about the system's physical interfaces. For example:

```
admin> dir  admin-state-phys
    17  08/06/1998 17:03:57  { shelf-1 slot-13 1 }
    17  08/06/1998 17:03:57  { shelf-1 slot-13 2 }
    17  08/06/1998 17:03:57  { shelf-1 slot-13 3 }
    17  08/06/1998 17:03:57  { shelf-1 slot-13 4 }
    17  08/06/1998 17:03:57  { shelf-1 slot-13 5 }
    17  08/06/1998 17:03:57  { shelf-1 slot-13 6 }
```

The system creates a profile for each of its physical interfaces. The Admin-State-Phys-If profile contains the following parameters (shown here with sample values):

```
[in ADMIN-STATE-PHYS-IF/{ shelf-1 slot-13 1 }]
device-address* = { shelf-1 slot-13 1 }
slot-type = hdlc2-card
snmp-interface = 0
modem-table-index = 0
desired-state = admin-state-up
desired-trap-state = trap-state-enabled
```

Parameter	Specifies
Device-Address	Physical slot address within the system.
Slot-Type	Type of card at that address.
SNMP-Interface	Interface table index assigned to the device whose state is stored in this profile.The system assigns a numeric value, which does not change as long as the interface is present in the system. If the card is removed and its profiles deleted (for example, by using a Slot –r command), the index number is freed for future use.
Modem-Table-Index	Modem table index assigned to the device whose state is stored in this profile. The system assigns a numeric value. The value is 0 for devices that are not modems.
Desired-State	Desired administrative state of the addressed device. The system sets it to Admin-State-Down if an operator downs the device, or to Admin-State-Up if an operator attempts to bring up the device in normal operations mode. An operator can change the admin state by using SNMP SET commands, or the Slot or If-Admin commands.
Desired-Trap-Sate	Desired link up/down enable state of the interface.The system sets it to Trap-State-Enabled if an operator specifies that linkUp/linkDown traps should be generated for the interface, or to Trap-State-Disabled if an operator specifies that linkUp/linkDown traps should not be generated for the interface.

Using the Device-State profile

Every host interface (such as an HDLC channel or a digital modem) or network interface (such as T1 or E1 channel) on the MAX TNT has a Device-State profile, which stores the current state of the device and allows you to change it. For example, each eight port T1 card has 192 Device-State profiles (one for each T1 channel). Similarly, each modem card has 48 Device-State profiles (one for each modem).

To open one of the profiles, proceed as in the following example:

```
admin> read device {{1 3 1} 24}
DEVICE-STATE/{ { shelf-1 slot-3 1 } 24 } read
admin> list
device-address* = { { shelf-1 slot-3 1 } 24 }
device-state = down-dev-state
up-status = idle-up-status
reqd-state = up-reqd-state
```

In the output, the Device-State parameter shows the current operational state of the device, which can be down, up, or none. (None indicates that the device does not exist.)

The Up-Status parameter is ignored unless the device is up (Device-State=Up-Dev-State). If the device is up, Up-Status shows the status of the device, which can be idle, reserved (will not be used until all idle devices of the same type are in use), or assigned (in use).

The Reqd-State parameter indicates the required operational state of the device, which can be up or down. Changing this value initiates a state change for the device. The change is complete when Device-State changes to match Reqd-State. This setting is not persistent across system resets or power cycles. At system startup, the MAX TNT reinitializes the required state to match the actual state of the card.

Using the Device-Summary profile

The read-only Device-Summary profiles record the status and availability of the modem and HDLC resources on the MAX TNT. This profile is not stored in NVRAM, so it is not persistent across system resets or power cycles.

To view the modem resources on a MAX TNT, proceed in the following example:

```
admin> read device-summary modem
DEVICE-SUMMARY/modem read
admin> list
[in DEVICE-SUMMARY/modem]
device-class* = modem
total-count = 48
operational-count = 48
disabled-count = 0
```

To view the HDLC resources on a MAX TNT, proceed as follows:

```
admin> read device-summary hdlc
DEVICE-SUMMARY/hdlc read
```

```
admin> list
[in DEVICE-SUMMARY/hdlc]
device-class* = hdlc
total-count = 96
operational-count = 96
disabled-count = 0
```

The parameters in the Device-Summary profiles are described below:

Parameter	Description
Device-Class	The type of device. Values can be any of the following: • Modem • HDLC • Unknown
Total-Count	Total number of devices in the specified class.
Operational-Count	Total number of devices in the specified class that are in the Up operational and Up administrative states.
Disabed-Count	Total number of devices in the specified class that are in the Down operational or Down administrative state.

Using the Slot-Info profile

The read-only Slot-Info profile stores information about each slot card that has successfully booted. This profile is not stored in NVRAM, so it is not persistent across system resets or power cycles. It is created when the slot card boots, and is deleted when the slot card is removed or when the MAX TNT system is rebooted. It can be read by SNMP managers.

To view the Slot-Info profile, read and list its contents, as in the following example:

```
admin> read slot-info {1 1 0}
SLOT-INFO/{ shelf-1 slot-1 0 } read
admin> list
[in SLOT-INFO/{ shelf-1 slot-1 0 }]
slot-address* = { shelf-1 slot-1 0 }
serial-number = 7470634
software-version = 7.0
software-revision = 4
software-level = b
hardware-level = 0
software-release = 1
```

For information about the parameters in the Slot-Info profiles, see the *MAX TNT Reference Guide* .

Using Slot-State profiles

When you set the required operational state of a slot, the MAX TNT initiates a state change. In terms of settings, Current-State changes to match Reqd-State. This setting is not persistent across system resets or power cycles. At system startup, the MAX TNT reinitializes the required state to match the actual state of the card.

To read a Slot-State profile and display its contents, proceed as in the following example:

```
admin> read slot-state {1 1 0}
SLOT-STATE/{ shelf-1 slot-1 1 } read
admin> list
slot-address* = { shelf-1 slot-1 0 }
current-state = oper-state-down
reqd-state = reqd-state-up
```

The slot address is the physical address of the slot, and cannot be set directly. The Current-State value shows the current operational state of the slot, and can be any of the states described below.

State	Description
Oper-State-Down	The slot is in a nonoperational state.
Oper-State-Up	The slot is in normal operations mode.
Oper-State-Diag	The slot is in diagnostics mode.
Oper-State-Dump	The slot is dumping core.
Oper-State-Pend	The slot is no longer down, but is not yet ready for normal operation. This value denotes a transitional state in which additional shelf-to-slot communications are required to make the slot fully operational.
Oper-State-Post	The slot is running a self-test.
Oper-State-None	The slot is empty.

The Reqd-State parameter indicates the required operational state of the slot, which can be up or down. Changing this value initiates a state change for the device. To use the Slot-State profile to change slot states, proceed as in the following example.

To bring a slot down:

```
admin> read slot-state {1 3 6}
SLOT-STATE/{ shelf-1 slot-3 6 } read
admin> set reqd-state = reqd-state-down
admin> write
SLOT-STATE/{shelf-1 slot-3 6} written
```

To bring the slot back up:

```
admin> set reqd-state = reqd-state-up
admin> write
SLOT-STATE/{ shelf-1 slot-3 6} written
```

Using ADSL profiles

The ADSL-CAP-Stat profiles and the ADSL-DMT-Stat profiles facilitate the administration of ADSL CAP and ADSL DMT connections, respectively.

Using the ADSL-CAP-Stat profile

The ADSL-CAP-Stat profile displays the status of the ADSL-CAP connection. Each port on the ADSL card has a separate profile.

To display the status of the ADSL lines, read and list the ADSL-CAP-Stat profile, as in the following example:

```
admin> read adsl-cap-stat {1 3 2}
ADSL-CAP-STAT/{ shelf-1 slot-3 2 } read

admin> list
[in ADSL-CAP-STAT/{ shelf-1 slot-3 2 }]
physical-address* = { shelf-1 slot-3 2 }
line-state = active
error-count = 0
physical-status = { 0 coe startup-handshake 0 0 232 0 1 auto auto +}
physical-statistic = { { 0 0 0 } no 57 0 passed 0 0 0 2 0 0 2 min+}
```

For descriptions of the parameters in the ADSL-CAP-Stat profile, see the *MAX TNT Reference Guide*.

Using the Physical-Status subprofile

The Physical-Status profile indicates the status of the ADSL interface. To display ADSL line status, list the Physical-Status subprofile, as in the following example:

```
admin> list physical-status
[in ADSL-CAP-STAT/{ shelf-1 slot-3 2 }:physical-status]
if-group-index = 0
unit-type = coe
dev-line-state = startup-handshake
up-stream-rate = 0
down-stream-rate = 0
major-firmware-ver = 232
minor-firmware-ver = 0
hardware-ver = 1
up-stream-constellation = auto
down-stream-constellation = auto
down-stream-operational-baud = 340
```

Following are the parameters in the ADSL-CAP-Status profile, with descriptions of what they indicate. (For complete descriptions, see the *MAX TNT Reference Guide*.)

Parameter	Indicates
IF-Group-Index	SNMP interface group index assigned to this port.

Parameter	Indicates
Unit-Type	Operating mode of the card. The values can be either of the following: • COE (Central Office Equipment) • CPE (Customer Premise Equipment)
Dev-Line-State	Interface status, as one of the following values: • Port-Up—The ADSL connection is operating normally and data can be transferred between nodes. • Test—The unit is undergoing power-on self-testing. • Startup-Handshake—The ADSL units are trying to establish a connection. This node is waiting for the remote node's connection request. If this condition persists, it could indicate that the connection between the units is faulty. • Startup-Training—The units are negotiating a connection. • Startup-Download—The unit is downloading firmware code into the ADSL card. • Idle—The unit has been reset and has not yet downloaded any code. • Down—The ADSL port is down. Data cannot be transmitted between nodes. The link goes down if one of the nodes loses power or if line quality degrades. (For an explanation of how the MAX TNT determines line quality, see "Connection-SQ (ADSL-CAP only)" on page 9-12. • Out-of-Service—The port has been administratively disabled.
Up-Stream-Rate	Upstream data rate. A value of 0 (zero) indicates that the data rate is unknown. RADSL ensures maximum throughput for any given line conditions. The better the line quality the higher the data rate.
Down-Stream-Rate	Downstream data rate. A value of 0 (zero) indicates that the data rate is unknown. RADSL ensures maximum throughput for any given line conditions. The better the line quality the higher the data rate. (For information on configuring the maximum downstream rate, see the *MAX TNT Hardware Installation Guide*.)
Major-Firmware-Ver	Major version number of the card's firmware.
Minor-Firmware-Ver	Minor version number of the card's firmware.
Hardware-Ver	Hardware version of the card.

Parameter	Indicates
Up-Stream-Constellation (ADSL-CAP only)	Operational upstream constellation.This correlates to rate. Constellation is the number of points within the digital spectrum. A value of 0 (zero) indicates that the upstream constellation is unknown. A value of `auto` indicates automatic.
Down-Stream-Constellation (ADSL-CAP only)	Operational downstream constellation.This correlates to rate. Constellation is the number of points within the digital spectrum. A value of 0 (zero) indicates that the downstream constellation is unknown. A value of `auto` indicates automatic.
Down-Stream-Operational-Baud (ADSL-CAP only)	Downstream operational baud rate.

Using the Physical-Statistic profile

The Physical-Statistic subprofile indicates the status of the ADSL interface. To display ADSL line statistics, list the Physical-Statistic profile, as in the following example:

```
admin> list physical-statistic
[in ADSL-CAP-STAT/{ shelf-1 slot-3 2 }:physical-statistic]
line-up-timer = { 0 0 0 }
rx-signal-present = no
line-quality = 57
up-dwn-cntr = 0
self-test = passed
rs-errors = 0
rs-corrected-errors = 0
transmit-power = 0
rx-attenuation = 2
connection-sq = 0
hdlc-rx-crc-error-cnt = 0
bert-timer = 2 minutes
bert-enable = no
bert-operation-state = stopped
bert-error-counter = 0
```

Following are brief descriptions of the parameters in the Physical-Statistic subprofile. (For complete descriptions, see the *MAX TNT Reference Guide*.)

Parameter	Indicates
Line-Up-Timer	Indicates the period of time the line has been in the up state, in the format {*dd hh mm*} where *dd* is the number of days, *hh* the number of hours, and *mm* the number of minutes.
RX-Signal-Present	Indicates whether this node is receiving a signal from the remote node.
Line-Quality	Indicates the line quality in decibels. For an ADSL interface, a reading of 18 dB or greater is required for reliable data transfer.

Parameter	Indicates
Up-Down-Cntr	Indicates the number of times the link has transitioned from an up state to a down state since the card was last reset.
Self-Test	Indicates whether the card has passed Power On Self Test (POST). Possible values are: • Passed • Failed
RS-Errors (ADSL-CAP only)	Indicates the Reed Solomon errors that have not been corrected. This value is only used by the CPE unit. If the CPE unit detects a very high rate of RS Errors (255 every 50ms) for 8 consecutive seconds, it disconnects the line.
RS-Corrected-Errors (ADSL-CAP only)	Indicates the number of Reed Solomon corrected errors.
Transmit-Power (ADSL-CAP only)	Indicates the transmission power level in dB.
RX-Attenuation (ADSL-CAP only)	Indicates the attentuation level of the receive signal from the far end.
Connection-SQ (ADSL-CAP only)	Indicates the signal quality (SQ) in dB. When the line is active, the MAX TNT compares Connection-SQ to Line-Quality. If the difference between the Line-Quality and the Connection-SQ readings is greater than 6dB for 22 seconds, the MAX TNT disconnects the line. This typically occurs when a line becomes open or the remote unit loses power.
HDSL-RX-CRC-Error-Cnt	Indicates the number of HDLC receive CRC errors.
BERT-Timer (ADSL-CAP only)	Specifies the BER test timer. If the two ends or the connection are not connected, the BERT-Timer does not apply. In this case, you must set BERT-Enable to No to end the BER test.
BERT-Enable (ADSL-CAP only)	Enables or disables the Bit Error Rate (BER) test. The BER test counts data errors that occur on each channel, to check the data integrity of the connection. If the two ends of the ADSL connection are physically connected, the BER test is run between the two units. If the two ends are not connected, the BER test is run within the card itself. Note that both ends of the connection must enable the BER test. A BER test interrupts normal data transmission.

Parameter	Indicates
BERT-Operation-State (ADSL-CAP only)	Indicates the state of the BER test. Possible values, and the status the indicate, are: • Waiting-for-511-sync—The node is waiting for the other node before starting its BER test. • Local-Loop-Active—The interface is in local analog loopback and is running the BER test. During an analog loopback, the card itself is looped back. No remote device is involved. • Active—The node is connected to the remote node and the BER test is running. • Stopped—The BER test is not enabled. • Loop-Back-Setup—The interface is being placed into analog loopback. During an analog loopback, the card itself is looped back. No remote device is involved. • Start-Up—The BER test is starting up. • Data-Overflow—During the test, a sudden surge of errors was received, causing the ADSL buffers to overflow. This could occur if the remote end stopped the test. • 511-Sync-Loss—The BER test went out of snyc.
BERT-Error-Counter (ADSL-CAP only)	Indicates the number of errors received during the BER test.

Using the ADSL-DMT-Stat profile

The ADSL-DMT-Stat profile displays the status of the ADSL-DMT connection. Each port on the ADSL card has a separate profile.

To display the status of the ADSL lines, read and list the ADSL-DMT-Stat profile, as in the following example:

```
admin> read adsl-dmt-stat {1 9 1}
ADSL-DMT-STAT/{ shelf-1 slot-9 1 } read

admin> list
[in ADSL-DMT-STAT/{ shelf-1 slot-9 1 }]
physical-address* = { shelf-1 slot-9 1 }
line-state = disabled
error-count = 0
physical-status = { 0 coe act-request 0 0 2 12 1 }
physical-statistic = { { 0 0 0 } no 6 passed 0 0 0 0 0 0 0 }
```

For descriptions of the parameters in the ADSL-CAP-Stat profile, see the *MAX TNT Reference Guide*.

Using the Physical-Status subprofile

The Physical-Status profile indicates the status of the ADSL-DMT interface. To display ADSL line status, list the Physical-Status subprofile, as in the following example:

```
admin> list physical-status
[in ADSL-DMT-STAT/{ shelf-1 slot-9 1 }:physical-status]
if-group-index = 0
unit-type = coe
dev-line-state = act-request
up-stream-rate = 0
down-stream-rate = 0
major-firmware-ver = 2
minor-firmware-ver = 12
hardware-ver = 1
```

For descriptions of these parameters, see “Using the Physical-Status subprofile” on page 9-9 or the *MAX TNT Reference Guide*.

Using the Physical-Statistic profile

The Physical-Statistic subprofile indicates the status of the ADSL interface. To display ADSL line statistics, list the Physical-Statistic profile, as in the following example:

```
admin> list physical-statistic
[in ADSL-DMT-STAT/{ shelf-1 slot-9 1 }:physical-statistic]
line-up-timer = { 0 0 0 }
rx-signal-present = no
up-dwn-cntr = 0
self-test = passed
hdlc-crc-errors = 0
near-end-tot-ES = 0
near-end-curr-atten = 0
near-end-curr-snr = 0
far-end-tot-ES = 0
far-end-curr-atten = 0
far-end-curr-snr = 0
```

Most of the parameters in the ADSL-DMT-Stat Physical-Statistic subprofile are the same as those in the ADSL-CAP-Stat Physical-Statistic subprofile. For descriptions of these parameters, see “Using the Physical-Statistic profile” on page 9-11 or the *MAX TNT Reference Guide*. The following parameters are specific to ADSL-DMT card.

Parameter	Description
Near-End-Tot-ES	Number of errored seconds on the link as measured by the COE device.
Near-End-Curr-Atten	Near-end line attenuation in dB.
Near-End-Curr-SNR	Near-end line quality (in dB) relative to a standard T1 line quality with an error rate of 10^{-7} errors per second.
Far-End-Tot-ES	Number of errored seconds on the link as measured by the CPE device.

Parameter	Description
Far-End-Curr-Atten	Far-end line attenuation in dB.
Far-End-Curr-SNR	Far-end line quality (in dB) relative to a standard T1 line quality with an error rate of 10^{-7} errors per second.

For complete information on these errors, see ANSI T1.413 issue 2.

Using DS3-ATM-Stat profiles

To display the status of the ATM DS3 line, read and list the DS3-ATM-Stat profile, as in the following example:

```
admin> read ds3-atm-stat {1 7 1}
DS3-ATM-STAT/{ shelf-1 slot-7 1 } read

admin> list
physical-address*={shelf-1 slot-7 1 }
line-state = active
f-bit-error-count = 0
p-bit-error-count = 0
cp-bit-error-count = 0
feb-error-count = 0
bpv-error-count = 0
loss-of-signal = False
loss-of-frame = False
yellow-receive = False
ais-receive = False
```

The Line-State parameter shows the overall state of the line which can be any of the following:

State	Description
Does-Not-Exist	Link is not physically on board.
Disabled	Line disabled.
Loss-of-Signal	Near end has lost signal.
Loss-of-Frame	Near end has lost frame.
Yellow-Alarm	Receiving yellow-alarm from far end.
AIS-Receive	Receiving alarm indication signal.
Active	Multipoint established.

The remaining parameters indicate the errors on the DS3 line. (Refer to RFC 1407 for complete description of these errors.)

Parameter	Description
F-Bit-Error-Count	Framing bit errors received since the last MAX TNT reset.

Parameter	Description
P-Bit-Error-Count	P-bit errors indicate that MAX TNT received a P-bit code on the DS3 M-frame that differs from the locally calculated code.
CP-Bit-Error-Count	For C-Bit-Parity lines indicates that number of parity errors since the last MAX TNT reset.
FEB-Error-Count	Far end block errors received since the last MAX TNT reset.
BPV-Error-Count	Bipolar Violation (BPV) errors may indicate that the line sent consecutive one bits with the same polarity. It could also mean that three or more consecutive zeroes were sent or an incorrect polarity.
Loss-of-Signal	True indicates a loss of signal. False indicates that the carrier is maintaining a connection.
Loss-of-Frame	True indicates a loss of framing. False indicates that the line is up and in frame.
Yellow-Receive	True indicates that the local device has received a Yellow Alarm indication. False specifies that the local device has not received a Yellow Alarm indication.
AIS-Receive	True indicates that the local device has received alarm indication signal. False indicates local device has not received and alarm indication signal.

Using IDSL-Stat profiles

To display the status of the IDSL line, read and list the IDSL-Stat profile, as in the following example:

```
admin> read idsl-stat {1 9 2}
IDSL-STAT/{ shelf-1 slot-9 2 } read

admin> list
[in IDSL-STAT/{ shelf-1 slot-9 2 }]
physical-address* = { shelf-1 slot-9 2 }
line-state = point-to-point
channel-state = [ nailed-up nailed-up ]
error-count = [ 0 0 ]
```

Following are the parameters in the IDSL-Stat profile, with brief descriptions of what they indicate. (For complete descriptions, see the *MAX TNT Reference Guide*.)

Parameter	Indicates
Physical-Address	Physical address of the IDSL card, in the format {*shelf slot item*} where *item* is the port number of the card.

Parameter	Indicates
Line-State	State of the line, as one of the following values: • Does-Not-Exist—The line is not physically connected. • Disabled—Line disabled. • No-Physical—No physical link. • No-Logical—Logical link failure. • No-Mgmt—Layer-2 link established but management entities not initialized. • Point-to-Point—Point-to-point link established. • Multipoint-1—Multipoint link established with one phone number and one SPID. • Multipoint-2—Multipoint link established with two phone numbers and two SPIDs.
Channel-State	State of the channels, as one of the following values: • Unavailable—Not available. • Unused—Not in use. • Out-of-Service—Out of service. • Nailed-Up—Nailed up. • Idle—Idle. • Clear-Pending—Clear pending. • Dialing—Dialing. • Ringing—Ringing. • Connected—Connected. • Signaling—Channel is a D Channel used for signalling. • Cut-through—Channel is a drop-and-insert source or destination. • Current-D—Channel is a current D Channel (NFAS signaling only). • Backup-D—Channel is a backup D Channel (NFAS signaling only). • Maintenance—Maintenance state.
Error-Count	Number of errors per channel that have occurred since the last reset.

Using SDSL profiles

SDSL profiles include SDSL and SDSL-Stat.

Using the SDSL Stat profile

The SDSL-Stat profile indicates the status of the SDSL connection. Each port on the SDSL card has a separate profile.

To display the status of an SDSL line, read and lists its SDSL-Stat profile, as in the following example:

```
admin> read sdsl-stat {1 1 2}
SDSL-STAT/{ shelf-1 slot-1 2 } read

admin> list

[in SDSL-STAT/{ shelf-1 slot-1 2 }]
physical-address* = { shelf-1 slot-1 2 }
line-state = active
error-count = 0
physical-status = { 0 coe port-up 784000 784000 13 2 2 }
physical-statistic = { { 1 13 55 } yes 15 1 passed 4 no +}
```

For descriptions of the parameters in the SDSL-Stat profile, see to the *MAX TNT Reference Guide*.

Using the Physical-Statistics subprofile

The Physical-Statistic subprofile indicates the status of the SDSL interface. To display SDSL line statistics, list the Physical-Statistic profile, as in the following example:

```
admin> list physical-statistic
[in SDSL-STAT/{ shelf-1 slot-1 2 }:physical-statistic]
line-up-timer = { 1 13 55 }
rx-signal-present = yes
line-quality = 15
up-dwn-cntr = 0
self-test = passed
far-end-db-attenuation = 4
firmware-startup-stage = normal-operation
hdlc-rx-crc-error-cnt = 5
bert-timer = 2 minutes
bert-enable = no
bert-operation-state = stopped
bert-error-counter = 0
```

Following are the parameters in the SDSL-Statistic subprofile, with brief descriptions of what they indicate. (For complete descriptions, see to the *MAX TNT Reference Guide*.)

Parameter	Indicates
Line-Up-Timer	Period of time the line has been in the up state, in the format {*dd hh mm*} where *dd* is the number of days, *hh* the number of hours, and *mm* the number of minutes.
RX-Signal-Present	The local node is/is not receiving a signal from the remote node.
Line-Quality	Line quality in decibels. For an SDSL interface, a reading of -5dB or better is required for reliable data transfer.
Up-Down-Cntr	Number of times the link has transitioned from an up state to a down state since the card was last reset.
Self-Test	Whether the card has passed or failed Power On Self Test (POST).
Far-End-Db-Attenuation	Attenuation level of the signal received from the far end.
Firmware-Startup-Stage	Current firmware state.
HDLC-RX-CRC-Error-Cnt	The number of CRC errors that have occurred. A few CRC errors are normal, but the line is disconnected if 1500 errors occur within a 2-second time period.
BERT-Timer	Specifies the BER test timer. If the two ends or the connection are not connected, the BERT-Timer does not apply. In this case, you must set BERT-Enable to No to end the BER test.
BERT-Enable	Enables or disables the Bit Error Rate (BER) test. The BER test counts data errors that occur on each channel, to check the data integrity of the connection. If the two ends of the ADSL connection are physically connected, the BER test is run between the two units. If the two ends are not connected, the BER test is run within the card itself. Note that both ends of the connection must enable the BER test. A BER test interrupts normal data transmission.

Parameter	Indicates
BERT-Operation-State	Indicates the state of the BER test. Possible values, and the status the indicate, are: • Waiting-for-511-sync—The node is waiting for the other node before starting its BER test. • Local-Loop-Active—The interface is in local analog loopback and is running the BER test. During an analog loopback, the card itself is looped back. No remote device is involved. • Active—The node is connected to the remote node and the BER test is running. • Stopped—The BER test is not enabled. • Loop-Back-Setup—The interface is being placed into analog loopback. During an analog loopback, the card itself is looped back. No remote device is involved. • Start-Up—The BER test is starting up. • Data-Overflow—During the test, a sudden surge of errors was received, causing the ADSL buffers to overflow. This could occur if the remote end stopped the test. • 511-Sync-Loss—The BER test went out of snyc.
BERT-Error-Counter	Indicates the number of errors received during the BER test.

Using the Phycial-Status subprofile

The Physical-Status profile indicates the status of the SDSL interface. To display SDSL line status, list the Physical-Status subprofile, as in the following example:

```
admin> list physical-status
[in SDSL-STAT/{ shelf-1 slot-1 2 }:physical-status]
if-group-index = 0
unit-type = coe
dev-line-state = port-up
up-stream-rate = 784000
down-stream-rate = 784000
major-firmware-ver = 13
minor-firmware-ver = 2
hardware-ver = 2
```

Following are the parameters in the SDSL-Status subprofile, with brief descriptions of what they indicate. (For complete descriptions, see to the *MAX TNT Reference Guide*.)

Parameter	Indicates
IF-Group-Index	SNMP interface group index assigned to this port.
Unit-Type	Operating mode of the card. The values can be either of the following. • COE (Central Office Equipment) • CPE (Customer Premise Equipment)

Parameter	Indicates
Dev-Line-State	Interface status. as one of the following values: • Config—The physical interface is being configured. • Deactivate—The interface is transitioning to a port down state. • Deactivate-lost—The interface is waiting for the Loss of Signal (LOS) Timer to expire. • Inactive—The interface is starting up. • Activating—The interface is waiting for the remote side to start up. • Active-RX—The interface is waiting for the remote side to start 4 level transmission. • Port-Up—The interface is connected to the remote node and data can be transferred. • Portup-Pending-Deactive—The interface experienced an LOS or noise-margin error. This occurs when the line detects noise of more than -5dB.
Up-Stream-Rate	Upstream data rate.
Down-Stream-Rate	Downstream data rate.
Major-Firmware-Ver	Major version number of the card's firmware.
Minor-Firmware-Ver	Minor version number of the card's firmware.
Hardware-Ver	Hardware version of the card.

Using SWAN-Stat profiles

A SWAN-Stat profile displays the status of a serial WAN line. Each line on the SWAN card has a separate profile.

To display the status of a serial WAN line, use the command, as in the following example:

```
admin> read swan-stat {1 8 1}
SWAN-STAT/{ shelf-1 slot-8 1 } read
admin> list
physical-address* = { shelf-1 slot-15 3 }
line-state = active
error-count = 0
```

For descriptions of the parameters in the SWAN-Stat profile, see to the *MAX TNT Reference Guide*.

Using T1-Stat profiles

The T1-Stat profile displays the status of the T1 lines and their channels. Each T1 line has a separate profile. When the T3 card is operational, it creates a T3-Stat profile and twenty-eight T1-Stat profiles, which store the current status of the DS3 and each component DS1. Similarly, when a FrameLine card is operational, it creates a 10 T1-Stat profiles, which store the current status each component DS1.

To display the status of the T1 line, read and list the T1-Stat profile, as in the following example:

```
admin> read t1-stat {1 8 1}
T1-STAT/{ shelf-1 slot-8 1 } read

admin> list
physical-address* = { shelf-1 slot-10 7 }
line-state = disabled
channel-state = [ disabled disabled disabled disabled disabled +
error-count = [ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ]
loss-of-carrier = False
loss-of-sync = False
ais-receive = False
yellow-receive = False
ber-receive = False
carrier-established = False
network-loopback = False
```

The Line-State parameter shows the overall state of the line which can be any of the following:

State	Description
Does-Not-Exist	Link is not physically on board.
Disabled	Line disabled.
Loss-of-Sync	Red-alarm state, plus or minus.
Yellow-Alarm	Yellow-alarm state.
AIS-Receive	Receiving keep-alive signal.
No-D-Channel	D-Channel failure.
Active	Multipoint established.

The channel-state parameter shows the state of each channel. Possible states are:.

State	Description
Unavailable	Not available.
Unused	Not in use.
Out-of-service	Out of service.
Nailed-up	Nailed.

The Error-Count parameter shows an error count for each channel.

For complete descriptions of the parameters in the T1-Stat profile, see to the *MAX TNT Reference Guide*.

Using UDS3-Stat profiles

To display the status of the UDS3 line, read and list the UDS3-Stat profile, as in the following example:

```
admin> read uds3-stat {1 13 1}
UDS3-STAT/{ shelf-1 slot-13 1 } read

admin> list
line-state = active
f-bit-error-count = 0
p-bit-error-count = 0
cp-bit-error-count = 0
feb-error-count = 0
bpv-error-count = 0
loss-of-signal = False
loss-of-frame = False
yellow-receive = False
ais-receive = False
```

The Line-State parameter shows the overall state of the line which can be any of the following:

State	Description
Does-Not-Exist	Link is not physically on board.
Disabled	Line disabled.
Loss-of-Signal	Near end has lost signal.
Loss-of-Frame	Near end has lost frame (also known as a red alarm).
Yellow-Alarm	Receiving yellow-alarm from far end.
AIS-Receive	Receiving alarm indication signal.
Active	Multipoint established.

The remaining parameters indicate the errors on the DS3 line. (Refer to RFC 1407 for complete description of these errors.)

Parameter	Description
F-Bit-Error-Count	Framing bit errors received since the last MAX TNT reset.
P-Bit-Error-Count	P-bit errors indicate that MAX TNT received a P-bit code on the DS3 M-frame that differs from the locally calculated code.
CP-Bit-Error-Count	For C-Bit-Parity lines indicates that number of parity errors since the last MAX TNT reset.
FEB-Error-Count	Far end block errors received since the last MAX TNT reset.

Parameter	Description
BPV-Error-Count	Bipolar Violation (BPV) errors may indicate that the line sent consecutive one bits with the same polarity. It could also mean that three or more consecutive zeroes were sent or an incorrect polarity.
Loss-of-Signal	True indicates a loss of signal. False indicates that the carrier is maintaining a connection.
Loss-of-Frame	True indicates a loss of framing (also known as a red alarm). False indicates that the line is up and in frame.
Yellow-Receive	True indicates that the local device has received a Yellow Alarm indication. False specifies that the local device has not received a Yellow Alarm indication.
AIS-Receive	True indicates that the local device has received alarm indication signal. False indicates local device has not received and alarm indication signal.

Getting MAX TNT Core Dumps

A

This appendix contains the following sections:

What is a core dump?

A MAX TNT core dump is a snapshot of MAX TNT shelf controller or slot card memory. An Ascend representative might ask you to obtain a core dump to help diagnose a problem. To get a core dump from the MAX TNT, you must use the Coredump command on the MAX TNT and the Ascendump utility on a local UNIX workstation.

The Coredump command controls how the MAX TNT generates core dumps. Ascendump controls how the MAX TNT core dumps are written to disk. You can specify that the core dump be collected whenever there is a fatal error, or you can get the core dump at any time from the server running Ascendump or from the MAX TNT itself.

The core-dump server can be connected through any LAN or WAN interface, and may be multiple hops away. The only restriction is that the data path from a crashing shelf or card must pass through shelves or cards that are still alive. The only exception is that a crashing Ethernet card can dump through one of its own Ethernet ports.

The MAX TNT uses UDP to write core dumps over the Ethernet.

Caution: Do not use core dumps unless specifically requested to by an Ascend representative.

Before you begin

Before installing and using the Ascendump utility, make sure you:

- Are familiar with the UNIX shell and know how to change directories, get information about files, use FTP, start processes, check available disk space, and so on.
- Have a local UNIX workstation running Solaris, SUNOS, or BSDI UNIX. (To use core dump on other versions of UNIX, contact technical support.)
- Have a minimum of 16Mb free disk space on the core-dump server. Note, however, that more space might be required under certain circumstances, such as if you are core dumping the core from the 32M DRAM card.
- Have downloaded the appropriate version of Ascendump from the Ascend FTP server (`ftp.ascend.com/pub/Utilities/coredump`).
- Have installed it in the directory from which you want to run it, and have used the `chmod +x` command to make the file executable.
- Have `gzip` on the core dump server and in the root users search path if you want to store core dumps as compressed files. If you specify compressed core dumps (the default), and `gzip` is unavailable, the core dumps are stored uncompressed.

To obtain a core dump from the MAX TNT, both the daemon and the MAX TNT must be configured to allow it.

The Ascendump daemon

Ascendump has the following syntax:

ascendump [**-v -r -c -u -p**] [**-n** ***email-recipient***] [**-s** ***slot***] [**-d** ***directory***] [***host***]

Option	Explanation
-v	Verbose mode, report detailed progress of core dump. With this option, Ascendump stops after a single dump. Without this option, Ascendump will accept multiple core dumps.
-r	Reset the MAX TNT after the core dump. This is the default in daemon mode.
-c	Do not reset the box after the core dump. This is the default in client mode.
-p	Print diagnostics to the terminal screen instead of Syslog. By default the server mode uses Syslog and the client mode prints to the terminal. Running Ascendump with this option does not allow multiple core-dump clients to dump at the same time.
-s ***slot***	Dump the memory of the card in slot number ***slot***. Network traffic will be forwarded through the shelf controller. In a multishelf system, a slot card can only dump through its own shelf controller.
-u	Store files uncompressed. By default files are compressed with `gzip`.

Option	Explanation
-n *email-recipient*	Send an email notification to the specified email recipient. You can use this option more than once to designate multiple recipients. You can also use mail aliases.
-d *directory*	The directory path for writing the core dumps. The default is `/usr/ascendumps`.
host	The name of the MAX TNT from which to get the core dump. This is known as client mode.

Coredump command

The Coredump command's syntax provides the following valid entries:

coredump

coredump enable | local | remote [*server*]

coredump disable

coredump now

coredump trace

Syntax element	Description
coredump	With no option specified, reports the enabled or status of Coredump and the core-dump server, if any.
enable	Enables Coredump. If you do not specify a server, the core-dump server remains unchanged.
local	The most commonly used mode. In Local mode the Ascendump daemon listens for packets from the MAX TNT. The Ascendump daemon operates in server mode, and the MAX TNT core dump facility operates in client mode.
remote	Enables the Ascendump daemon to pull a core dump from the MAX TNT. Remotely initiated core dumps can be a security risk, so they are disabled by default. If you enable remote core dumps, they remain enabled only until the MAX TNT resets. That is, a reset restores the default setting.
server	The host that has the Ascendump daemon installed. If you do not specify a server, the MAX TNT uses the previously configured server. To specify that the broadcast address be used, enter a hyphen (-).
disable	Disables Coredump.
now	Forces an immediate core dump to the machine running the Ascendump daemon. This is useful for testing the core dump process.
trace	Toggles serial debug traces which can be useful to an Ascend representative if a customer is having difficulties.

Core dump naming conventions and file characteristics

The core-dump files use the following naming convention:

hostname-[*shelf, slot*]-*loadname-swversion-YYMMDD-HH:MM.gz*

where:

- *hostname* is the hostname or IP address of the Ascend unit.
- *shelf,slot* is the shelf and slot number of the card that has dumped its core. (This applies only to the MAX TNT.)
- *loadname* is the name of the software load running on the MAX TNT.
- *swversion* is the version of the software load running on the MAX TNT.
- *yymmdd-hh:mm* is a date and time stamp. Each dump file can be four to eight megabytes in size.

For example:

```
tnt10.abc.com-1,3-tntmdm56k-1.3Ap22-980101-13:42.gz
```

When transferring the core-dump files via FTP, use binary mode.

Trigger events

The events that normally trigger a core dump are system or slot-card resets. These usually show up in the fatal error log either as “Fatal Errors” or “Operator Resets.” You cannot specify the types of events that trigger core dumps.

UDP port numbers

The MAX TNT listens for core dumps on the UDP port given by the following formula:

10,000 + (*shelf-number* *100) + *slot-number*

For example, for a card on shelf 1, slot 5, the UDP port for the core dump is 10105. For the shelf controller (slot number 17) on shelf 1, the UDP port for the core dump is 10117. Similarly, the shelf controller on shelf 8 uses UDP port 10817.

Examples

This section uses examples to show how to get core dumps from the MAX TNT.

Enabling Ascendump

To start the Ascendump daemon, proceed as in the following example:

```
% ./ascendump -v -u -d /usr/ascendumps
```

This example runs the daemon in verbose mode and will write the core dumps in uncompressed format to `/usr/ascendumps`.

Enabling core dumps on the MAX TNT

In the following example, the MAX TNT writes the core dump to the host at 172.31.4.34 whenever there is a fatal error:

```
admin> coredump local 172.31.4.34
coreDump: Sending arp request...
core dump server is ‘172.31.4.34 ip=[172.31.4.34/16],
mac=[00:60:83:7d:15:8f]
coredump over UDP is enabled locally only with server
172.31.4.34
```

Pulling a core dump from the MAX TNT

In the following example, the MAX TNT enables the Ascendump daemon to solicit a dump from the MAX TNT. The Ascendump daemon is operating in client mode, and the MAX TNT core-dump facility is operating in server mode.

```
admin> coredump remote
```

Once remote core dumps are enabled on the MAX TNT, an administrator can “pull” a core dump as in the following example:

```
% ascendump -d /usr/ascendumps tnt10
```

where `/usr/ascendumps` is the directory on the Ascendump server and `tnt10` is the name of the MAX TNT from which to get the core dump.

Initiating an immediate core dump

In the next example, an administrator forces an immediate core dump:

```
admin> coredump now
```

Getting core dumps from slot cards

You can configure the Ascendump daemon to request a core dump from a particular MAX TNT slot. In the following example the modem card in slot 4 of the MAX TNT named `tnt10` will write to the Ascendump server when it crashes:

1. After opening a session with the card, execute Coredump with the remote option:

   ```
   modem-4> coredump remote
   ```

2. Start the Ascendump daemon in slot mode:

   ```
   % ./ascendump -v -u -s 4 -d /usr/ascendump
   ```

Disabling core dumps

To disable core dumps on the MAX TNT:

```
admin> coredump disable
coredump over UDP is disabled
```

Fatal error log and core dumps

The fatal-error log lists the pseudouser `coredump` as the responsible user when the master shelf controller resets after a core dump. For example:

```
OPERATOR RESET:  Index:  99  Revision: 1.3Ap8  Shelf 1 (tntsr)
        Date: 09/12/1997.       Time: 15:52:43
        Reset from unknown, user profile coredump.
```

Troubleshooting core dumps

Take the following steps if you have difficulty setting up the Ascend core dumps:

1. If you have previously installed Ascendump in `inetd.conf`, temporarily disable it now, by commenting out the Ascendump line, then, logged in as root, send the `SIGHUP` command to `inetd`.
2. Change to a writable directory, and enter **`ascendump -p -v -d`**
   - **`-v`** is verbose mode, which prints progress reports as the core dump proceeds, keeps the daemon in the foreground, and handles dumps serially, all of which make debugging easier.
   - **`-p`** prints diagnostics to `stderr` instead of through Syslog (whose output on most systems goes to `/var/adm/messages`).
   - **`-d`** puts the dump files in the current directory.

   Performing initial tests in this manner saves time by making failures immediately diagnosable.
3. On the MAX TNT, enable core dumps to the server machine that is running Ascendump.
4. Look for old debug profiles by entering, **`dir debug`** from the master shelf controller.

   The only reason to have a debug profile on a card other than the master shelf controller is to override the settings for the master shelf controller. Unless you want to do that, you should define a single debug profile for the master shelf controller and delete all other debug profiles. You can edit Debug profiles by using the Read, Set, and Write commands, or by using the **`coredump local`** ***`server`*** command, where ***`server`*** is the IP address of the core-dump server.
5. Test slot-card dumps by opening a session with a slot card. You should perform a test dump first on the T1 or E1 card, if present, because these cards have smaller memories, and are quick to reboot.
6. From the session on the card, enter **`coredump`** to check the status of core dump. The resulting output should report that core dump is enabled and that dumps will be directed to the server you specified in step 3.
7. Force a core dump with the following command:

   `coredump now`

   Ascendump should print something like this:

   ```
   $ ascendump -p -v -d

   ascendump: Dumping compressed DRAM image to ‘./tnt10.abc.com-1,11-
   tnt8t1-1.3Ae0-971022-11:17.gz’
   Section ‘.data’: dumping 2048 pages from address 0x80000000
   .......1 Mb.......2 Mb
   ```

Occasionally, core dump fails because `gzip` is not installed or not in the user's path. If this is the case, you should download `gzip-1.2.4.tar.gz` from any GNU FTP mirror site, then compile and install it, or use the `-u` (uncompressed) option in the Ascendump command line.

If you still have unexplained failures, run `tcpdump` or `snoop` or a packet sniffer on the Ethernet segment attached to the MAX TNT that is in the route to the dump server. Do the same on the Ethernet segment attached to the dump server in the route to the MAX TNT.

Ascend Coredump uses UDP, so filter UDP packets. If there's too much UDP traffic, you might want to filter on port-number ranges as well. For information about the UDP port core dump uses, see "UDP port numbers" on page A-4.

Proceed to testing more cards by opening CLI channels to them and using the `coredump now` command. Finish by testing Coredump from the master shelf controller.

Once you have established that core dump works, reinstate your `inetd.conf` entry, if present, or add one if necessary. Be sure that the entry points to the same Ascendump binary that you just tested.

Here is a sample `inetd.conf` entry:

```
ascendump dgram udp nowait root /usr/local/bin/ascendump ascendump -n
dump-notify
```

The `-n dump-notify` argument tells Ascendump to send email to the email alias `dump-notify` whenever a core dump is captured.

MAX TNT Log Messages

B

This chapter covers the following topics:

The MAX TNT logs fatal and warning error messages to the fatal error log. If the system crashes before creating a log entry, it prints a stack trace to the console serial port. System-status messages, however, go to the Syslog host (if enabled) and the Status log.

Fatal and warning error messages

Each time the MAX TNT reboots, it logs a fatal error message to the fatal error log. The fatal error log also notes Warnings, which indicate situations that did not cause the MAX TNT to reset. Development engineers use Warnings for troubleshooting purposes. When a Warning occurs, the MAX TNT has detected an error condition and has recovered from it. Available flash space limits the number of entries in the fatal error log, and entries rotate on a First-in, First-out (FIFO) basis. You can clear the log by using the Clr-History command.

Format of fatal and warning error messages

Fatal and warning messages have the format shown in the following example:

```
WARNING:  Index: 171  Revision: 2.0.0 Slot 9/2 (tnthdlc)
Date: 02/28/1998.       Time: 20:57:59
Location: e0020b54 e006f568 e005d6b8 e005fd90 e005e4dc e00770a8
```

The first line indicates the type of error (fatal or warning), the index number of the error, the software revision number, the shelf and slot on which the error occurred, and the software load. The fatal log from older software versions display shelf 0.

The second line shows the date and time of the error.

The third line displays the top six program counter addresses from the execution stack active at the time of the crash.

Definitions of fatal errors

Following are definitions, by index number, of the fatal errors that the MAX TNT can report: if you experience a fatal error, contact Ascend Technical Support.

Index	Definition
1	Assert invoked during program execution. An Assert has been placed in the code. This problem can be either hardware related or software related.
2	Out of memory during memory allocation This is an out-of-memory condition, sometimes termed a memory leak.
3	Bad profile (T1 DSL related)
4	Switch type bad
5	LIF error
6	LCD error
7	ISAC (BRI) timeout BRI physical layer timeout.
8	Processor exception A processor-exception error caused the reset.
9	Invalid task switch (EXEC)
10	No mail descriptor (EXEC) This reset occurs if the MAX TNT tries to allocate a mail message when there are none left. The cause is usually a memory leak.
11	No mail buffer memory (EXEC)
12	No task to run (EXEC)
13	No timer memory (EXEC)
14	No timer pool (EXEC)
15	Wait called while in critical section (EXEC)
16	DSP not responding
17	DSP protocol error
18	DSP internal error
19	DSP loss of sync
20	DSP unused
21	DDD not responding
22	DDD protocol error
23	X25 buffer error
24	X25 init error

Index	Definition
25	X25 stack error
27	Memory allocation of zero length
28	Memory allocation of negative length
29	Task infinite loop The reset was the result of a software loop.
30	Too large memory copy
31	Magic sequence missing (MEMCPY)
32	Wrong magic sequence (MEMCPY)
33	Bad start address (MEMCPY)
34	IDEC timeout IDSL physical layer timeout.
35	EXEC restricted
36	Stack overflow
37	DRAM card error Indicates that a DRAM card of unknown size is inserted in the DRAM slot or that the DRAM card failed POST. Applies to the Pipeline 220 only.
39	No priority 2 task This error occurs if the MAX TNT has not run a priority 2 task in the last minute. Tasks in the MAX TNT are assigned priorities. Because the main routing task runs at priority 2, this error means that the MAX TNT has been hung for 1 minute.
40	Protection fault
99	Operator reset This reset is logged immediately before the MAX TNT goes down. Instead of a standard stack backtrace, the message includes the active security-profile index. 0 (zero) indicates an unknown security profile. On the MAX TNT, the Default profile is number 1, and the Full Access profile is number 9.
100	System up As a complement to entry 99, this entry is logged as the MAX TNT is coming up. For a normal, manual reset, you should see a fatal error 99 followed by a fatal error 100.

Definitions of warning messages

Warnings are not the results of reset conditions. Most are detected problems from which the MAX TNT typically recovers fully. Following are the definitions, by index number, of the warnings the MAX TNT can report. Warning messages, by themselves, are not necessarily cause for concern. They are used by development engineers to determine the cause of fatal errors. Contact technical support if warning messages are accompanied by fatal errors.

Index	Definition
101	Buffer already in use
102	Buffer belongs to wrong pool
103	Buffer belongs to wrong heap
104	Buffer not previously allocated
	This warning can be logged under different conditions. For example, double freeing of memory and low-memory conditions can both generate a warning 104.
105	Buffer bad memory allocation
106	Buffer belongs to bogus pool
107	Buffer belongs to bogus heap
	Memory management code (or other modules) detected that the buffer header of what should have been a free buffer was corrupted by the previous overwrite.
108	Buffer negative length memory allocation
	A negative length request was made to the memory allocation code.
109	Buffer zero length memory allocation
	This warning is similar to Warning 108, except that a zero length request is made to the memory allocation code.
110	Error in buffer boundary
111	Error buffer too big
	Indicates that a software routine has tried to allocate a block of memory greater than 64Kbytes.
112	Error buffer null
113	Error buffer segment count zero
114	Error buffer trailer magic
115	Error in buffer trailer
116	Error in buffer trailer length
117	Error in buffer trailer user magic
118	Error buffer write after free
119	Error buffer not in use
120	Error buffer magic in memory copy

Index	Definition
121	Error next buffer magic in memory copy
130	PPP async buffer in use
	Indicates a PPP error.
140	Error no timers
145	LCD memory allocation failure
	Indicates that a memory-copy routine was called, but the source buffer was much larger than expected.
150	Error memory copy too large
151	Error memory copy magic missing
152	Error memory copy wrong magic
153	Error memory copy bad start address
154	WAN buffer leak
	Indicates an error in the WAN drivers.
160	Error in terminal-server state
	Indicates an error in the WAN drivers.
161	Error in terminal server semaphore
165	Error in telnet free driver
170	STAC timeout
	Indicates a hardware error in the STAC compression chip.
171	STAC data not owned
	Error in the STAC compression chip.
175	EXEC failure
	Indicates that there is insufficient memory to start a new task.
176	EXEC restricted
177	EXEC no mailbox
178	EXEC no resources
179	Unexpected error
180	Channel map stuck
	Caused by a missing channel on a T1/PRI line.
181	Channel display stuck
182	New call without disconnect request
	Indicates that a Disconnect message to the Central Office (CO) was not sent. The problem can be caused by conditions on the MAX TNT or at the CO. When the MAX TNT encounters the condition, it assumes the CO is correct, and answers the call.

Index	Definition
183	New call without disconnect response
184	Disconnect request dropped
185	Spyder buffer error
186	Spyder descriptor error
190	TCP send buffer too big
191	TCP sequence gap
192	TCP too much data
193	TCP write attempt too large
194	TCP options bad
195	Modem message parsing failed
301	TACACS Plus pointer inconsistency
302	TACACS Plus index inconsistency
303	TACACS Plus TCP inconsistency
304	TACACS Plus TCP out-of-range socket
305	TACACS Plus socket mismatch
306	TACACS Plus unexpected authentication state
381	Error in filter list
382	Error no count in filter list
383	Error mismatch count filter list
550	No Ethernet transmit buffer
1001	Waiting for Ethernet controller
1002	Ethernet ACK command failed
1003	Ethernet reset invoked
1006	Ethernet controller unavailable (wait fail)
1010	Bad Ethernet transmit interrupt
1011	Ethernet transmit not completed

Fatal crash information on console

If the MAX TNT crashes without being able to write to the fatal error log, it prints a stack trace to the console serial port at the bit rate defined in the Serial profile. The trace reports the following information:

`FE: N, Load: loadname, Version: version`
`Stack trace: 0xaddr-0 0xaddr-1 0xaddr-2 0xaddr-3 0xaddr-4 0xaddr-5`

The first line indicates the number of the error and the software revision number.

The second line displays the top six program counter addresses from the execution stack active at the time of the crash.

Syslog messages

Syslog offloads to a host computer, known as the Syslog host. The Host parameter in the Log profile specifies the Syslog host, which saves the system status messages in a log file.

See the UNIX man pages about `logger(1)`, `syslog(3)`, `syslog.conf(5)`, and `syslogd(8)` for details of the `syslog` daemon. The Syslog function requires UDP port 514.

The MAX TNT can report the following session data about various errors logged via Syslog:

Data	Description
`[shelf/slot/line/channel]`	Physical channel identifier.
`[MBID xxx]`	Session identifier.
`[name]`	The authenticated name.
`[ calling -> called ]`	The calling number or the called number, or both.
`Progress code`	An Ascend-specific code indicating the progress of the call. (For a list of progress codes, see the *MAX TNT Reference Guide*.)
`Disconnect code`	An Ascend-specific code indicating the reason the call was disconnected. (For a list of disconnect codes, see the *MAX TNT Reference Guide*.)

For a given session identifier, multiple physical channel identifiers are possible. For example, one identifier might be for the T1 line, and another for the HDLC channel or modem number. This is shown in the sample log below, in which messages include the MBID, DNIS, and CLID in brackets. In this example, slot 1/2 is an 8T1 card, and slot 1/3 is a 48-modem card.

```
...: [1/2/1/2] [MBID 1; 9995551212 -> 7898] Incoming Call
...: [1/3/1/0] [MBID 1; 9995551212 -> 7898] Assigned to port
...: [1/2/1/2] [MBID 1; 9995551212 -> 7898] Call Connected
...: [1/3/1/0] [MBID 1] [johnc-pc] LAN session up: <johnc-pc>
...: [1/3/1/0] [MBID 1] [johnc-pc] LAN session down: <johnc-pc>
...: [1/3/1/0] [MBID 1; 9995551212 -> 7898] Call Terminated
...: [1/3/1/0] [MBID 1] [johnc-pc] : STOP: 'johnc-pc'; cause 45.;
progress 60.; host 10.1.26.2
```

End of call information

If the Call-Info parameter is set to End-of-Call, the MAX TNT reports the following information to Syslog at the end of each authenticated call:

- Station name
- Calling phone number
- Called phone number

- Encapsulation protocol
- Data rate (in bits per second)
- Progress code and disconnect reason
- Number of seconds before authentication
- Number of bytes or packets received during authentication
- Number of bytes or packets sent during authentication
- Length of session (in seconds)
- Number of bytes or packets received during the session
- Number of bytes or packets sent during the session

The following example of a Syslog message shows the information it provides about the terminated call:

```
"Conn=("cjones-p50" 5106785291->? PPP 56000 60/185) \
Auth=(3 347/12 332/13) \
Sess=(1 643/18 644/19), Terminated"
```

The information also appears in the connection-status window, and is logged as a message at level Info.

If some of the information is not available, that field displays either a question mark (for strings) or a zero (for numerals).

DNIS and CLID information

Syslog messages pertaining to a call display DNIS and CLID information, provided that the information is known. Following is an example that shows the DNIS 7895 in Syslog messages:

```
LOG info, Shelf 1, Controller, Time: 17:48:56--
† shelf 1, slot 1, line 1, channel 6, dnis 7895, Incoming Call, MBID
001

LOG info, Shelf 1, Controller, Time: 17:48:56--
† shelf 1, slot 2, dnis 7895, Assigned to port, MBID 001

LOG info, Shelf 1, Controller, Time: 17:48:57--
† shelf 1, slot 1, line 1, channel 6, dnis 7895, Call Connected, MBID
001

LOG warning, Shelf 1, Controller, Time: 17:49:20--
† shelf 1, slot 1, line 1, channel 6, dnis 7895, Call Disconnected

LOG info, Shelf 1, Controller, Time: 17:49:20--
† shelf 1, slot 2, Call Terminated
```

Syslog messages initiated by a Secure Access Firewall

Depending on the settings specified in Secure Access Manager (SAM), the MAX TNT might generate Syslog packets about packets detected by Secure Access Firewall. By default, SAM specifies generation of a Syslog message about every packet blocked by the firewall. All messages initiated by a firewall are in the following format:

date time router name ASCEND*: interface message*

- *date* is the date the message was logged by syslog.

- *time* is the time the message was logged by syslog.
- *router* is the router this message was sent from.
- *interface* is the name of the interface (ie0, wan0, and so on), unless a call filter logs the packet as it brings up the link, in which case the word call appears.
- The *message* format has a number of fields, one or more of which may be present.

The message fields appear in the following order:

protocol local direction remote length frag log tag

Table B-1. Syslog message fields for Secure Access Firewalls

Field	Description
protocol	Can be the four hexadecimal character Ether Type or one of the following network protocol names: ARP, RARP, IPX, Appletalk. For IP protocols, the field contains either the IP protocol number (up to 3 decimal digits) or one of the following names: IP-in-IP, TCP, ICMP, UDP, ESP, AH. In the special case of ICMP, the field also includes the ICMP Code and Type ([Code]/[Type]/icmp).
local	For non-IP packets, local is the source Ethernet MAC address of transmitted packets and the destination Ethernet MAC address of received packets. For a nonbridged WAN connection, the two MAC addresses are zeros. For IP protocols, local is the IP source address of transmitted packets and the IP destination address of received packets. In the case of TCP or UDP, it also includes the TCP or UDP port number ([IP-address];[port]).
direction	An arrow (<- or ->) indicating the direction in which the packet was traveling (receive and send, respectively).
remote	For non-IP protocols, remote has the same format that *local* has non-IP packets, but remote shows the destination Ethernet MAC address of transmitted packets and the source Ethernet MAC address of received packets. For IP protocols, remote has the same format as local but shows the IP destination address of transmitted packets and the IP source address of received packets.
length	The length of the packet in octets (8-bit bytes).
frag	Indicates that the packet has a nonzero IP offset or that the IP More-Fragments bit is set in the IP header.

Table B-1. Syslog message fields for Secure Access Firewalls (continued)

Field	Description
log	Reports one or more messages based upon the packet status or packet header flags. The packet status messages include: • corrupt—the packet is internally inconsistent • unreach—the packet was generated by an "unreach=" rule in the firewall • !pass—the packet was blocked by the data firewall • bringup—the packet matches the call firewall • !bringup—the packet did not match the call firewall • TCP flag bits that will be displayed include syn, fin, rst. • syn is will only be displayed for the initial packet which has the SYN flag and not the ACK flag set.
tag	contains any user defined tags specified in the filter template used by SAM.

The backoff queue error message in the Syslog file

Accounting records are kept until they are acknowledged by the accounting server. Up to 100 unacknowledged records are stored in the backoff queue. If the unit never receives an acknowledgment to an accounting request, it will eventually run out of memory. In order to keep this situation from the occurring, the unit deletes the accounting records and displays this error message in the syslog file:

```
Backoff Q full, discarding user username
```

This error generally occurs for one of the following reasons:

- You enabled RADIUS accounting on the MAX TNT, but not on the RADIUS server.
- The Acct-Port or Acct-Key are incorrect. The Acct-Key must match the value assigned in the RADIUS clients file or the TACACS+ configuration file.
- You are using the Livingston server instead of the Ascend server.

Flash card error messages

When a Load, Format, or Dircode command fails, the MAX TNT logs the messages described in this section.

Load command messages

Table B-2 lists the error messages that might appear when using the Load command:

Table B-2. Load command error messages

Error message	Description
`load: error: flash card write failed: card full`	There is no space to load software on the flash card.
`load: error: specified flash card not present`	No flash card is detected in the specified slot (1 or 2).
`load: error: specified flash card not formatted`	A Format command is required before loading the software.
`load: error: specified flash card has obsolete format`	The flash card was formatted for a MAX TNT 1.3x system. You must reformat the card to use it with release 2.0.0 or later. See the release notes for information about formatting a flash card for release 2.0.0 or later.
`load: error: specified flash card is write-protected`	The flash card's write-protect switch is set.
`load: error: specified flash image is currently in use`	A slot card in the LOAD state is currently accessing the flash card.

Format command messages

Table B-3 lists the error messages might appear when using the Format command:

Table B-3. Format command error messages

Error message	Description
`error: flash card` *N* `is not present`	No flash card is detected in the specified slot (1 or 2).
`error: flash card` *N* `is unavailable`	The flash card in the specified slot is already being formatted, is just coming up, or is in an error condition.
`error: flash card` *N* `is write-protected`	The write-protect switch is set on the card in the specified slot (1 or 2).

Table B-3. Format command error messages (continued)

Error message	Description
`error: flash card` *N* `is currently in use`	One or more images on the flash card are being read by a slot card in the LOAD state or are being written as part of a code download.

Dircode command messages

Table B-4 lists the error messages might appear when using the Dircode command:

Table B-4. Dircode command error messages

Error message	Description
`Card` *N* `is not formatted for use with this system`	The flash card is blank, corrupted, or formatted for another environment, such as DOS. To use this card, you must issue a Format command first.
`Card` *N* `is temporarily unavailable`	The flash card is currently coming up or is being formatted.
`Card` *N* `is unavailable`	The flash card experienced an error and is inaccessible.Check that the card is inserted properly.
`Card` *N* `uses a format which is no longer supported`	The flash card was formatted for a MAX TNT 1.3x system. You must reformat the card to use it with release 2.0.0 or later. See the release notes for information about formatting a flash card for release 2.0.0 or later.

PPP Decoding Primer

C

This appendix covers these topics:

Overview

Many of the diagnostic commands display raw data. This Primer is designed to assist you in decoding PPP, MP, MP+ and BACP negotiations. The negotiations can be logged with the Diagnostic commands `PPPDump`, `WANDisplay`, `WANDSess`, `WANNext` or `WANOpen`. For more detailed information than this guide provides, refer to specific RFCs. A partial list of pertinent RFCs appears at the end of this guide.

Breaking down the raw data

An important concept to keep in mind is that each device negotiates PPP independently, so the options might be identical for each direction of the session.

During PPP negotiation, frame formats in the various protocols are very similar. THey share the following characteristics:

- `FF 03` indicating it is a PPP frame.
- A two-byte Protocol Identifier.
- A one-byte Packet Format ID number
- A one-byte ID number.
- A two-byte length.
- Options for the protocol.

Below is a table of the most common protocols you'll see in Ascend diagnostic traces:

Identifier:	**Description:**
`C0 21`	Link Control Protocol (LCP)
`C0 23`	Password Authentication Protocol (PAP)

Identifier:	Description:
C2 23	Challenge Handshake Authentication Protocol (CHAP)
80 21	Internet Protocol (IP)
80 29	Appletalk Protocol
80 2B	Novell's Internetwork Packet Exchange (IPX)
80 31	Bridging PDU
80 FD	Compression Control Protocol (CCP)

Following are the packet formats:

Packet Format ID	Description
01	Configure Request
02	Configure Acknowledgment
03	Configure Non-Acknowledgment
04	Configure Reject
05	Terminate Request
06	Terminate Acknowledgment
07	Code Reject
08	Protocol Reject
09	Echo Request
0A	Echo Reply
0B	Discard Request

Note: If a packet received from the wan fails the Cyclic Redundancy Check (CRC) the display is similar to the following, where RBAD denotes Received BAD:

```
RBAD-27:: 8712 octets @ 26CFE8
[0000]: fe dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd
[0010]: dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd
[0020]: dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd
[0030]: dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd dd
```

Annotated Traces

Use the following traces as guides to help you decode other traces.

LCP Configure Request - MP+, MRU of 1524, MRRU of 1524 and End Point Discriminator using the device's MAC address:

```
XMIT-3:: 29 octets @ 2C2E94
[0000]: ff 03 c0 21 01 01 00 19 00 04 00 00 01 04 05 f4
[0010]: 11 04 05 f4 13 09 03 00 c0 7b 4c e0 4c
```

This is a second LCP Configure Request from the same device. Everything in the packet is identical to the previous packet, except the ID number has incremented from 01 to 02:

```
XMIT-3:: 29 octets @ 2C2E94
[0000]: ff 03 c0 21 01 02 00 19 00 04 00 00 01 04 05 f4
[0010]: 11 04 05 f4 13 09 03 00 c0 7b 4c e0 4c
```

LCP Configure Request - CHAP authentication, Magic number

```
RECV-3:: 19 octets @ 2BEB8C
[0000]: ff 03 c0 21 01 60 00 0f 03 05 c2 23 05 05 06 4e
[0010]: 36 c9 05
```

LCP Configure Acknowledgment - This device will authenticate using CHAP. The Magic number is also acknowledged:

```
XMIT-3:: 19 octets @ 2C2E94
[0000]: ff 03 c0 21 02 60 00 0f 03 05 c2 23 05 05 06 4e
[0010]: 36 c9 05
```

LCP Configure Reject - MP+, MRU of 1524, MRRU of 1524 and End Point Discriminator.

This rejection shows two things. It shows that the remote side does not support MP+ or MP, since MP+ and the MRRU were rejected. This will have to be a PPP connection. Also, since the MRU of 1524 was rejected, the default of 1500 is assumed. There needs to be an MRU, so a rejection of a given value only means to use the default value.

At this point, this device will need to retransmit another LCP Configure Request, removing all the rejected options.

```
RECV-3:: 29 octets @ 2BF1A4
[0000]: ff 03 c0 21 04 02 00 19 00 04 00 00 01 04 05 f4
[0010]: 11 04 05 f4 13 09 03 00 c0 7b 4c e0 4c
```

LCP Configure Request - Note all values that were previously rejected are no longer in the packet:

```
XMIT-3:: 8 octets @ 2C2E94
[0000]: ff 03 c0 21 01 04 00 04
```

LCP Configure Acknowledgment -

```
RECV-3:: 8 octets @ 2BF7BC
[0000]: ff 03 c0 21 02 04 00 04
```

At this point, since both sides have transmitted LCP Configure Acknowledgments, LCP is up and the negotiation moves to the authentication phase.

This device receives a CHAP challenge from the remote end:

```
RECV-3:: 21 octets @ 2BFDD4
[0000]: ff 03 c2 23 01 01 00 11 04 4e 36 c9 5e 63 6c 63
[0010]: 72 34 30 30 30
```

This device transmits its encrypted user name and password:

```
XMIT-3:: 36 octets @ 2C2E94
[0000]: ff 03 c2 23 02 01 00 20 10 49 b8 e8 54 76 3c 4a
[0010]: 6f 30 16 4e c0 6b 38 ed b9 4c 26 48 5f 53 65 61
[0020]: 74 74 6c 65
```

The remote device sends a CHAP Acknowledgment:

```
RECV-3:: 8 octets @ 2C03EC
[0000]: ff 03 c2 23 03 01 00 04
```

At this point, the negotiation moves from authentication to negotiation of Network Control Protocols (NCPs). Ascend supports Bridging Control Protocol (BCP), IPCP, IPXCP and ATCP.

IPCP Configure Request - Van Jacobsen Header Compression, IP address of 1.1.1.1

```
RECV-3:: 20 octets @ 2C0A04
[0000]: ff 03 80 21 01 e3 00 10 02 06 00 2d 0f 00 03 06
[0010]: 01 01 01 01
```

BCP Configure Request -

```
RECV-3:: 8 octets @ 2C101C
[0000]: ff 03 80 31 01 55 00 04
```

IPCP Configure Request - IP address of 2.2.2.2

```
XMIT-3:: 14 octets @ 2C2E94
[0000]: ff 03 80 21 01 01 00 0a 03 06 02 02 02 02
```

IPCP Configure Reject - Van Jacobsen Header Compression. The remote device should send another IPCP Configure Request and remove the request to do VJ Header Compression:

```
XMIT-3:: 14 octets @ 2C2E94
[0000]: ff 03 80 21 04 e3 00 0a 02 06 00 2d 0f 00
```

BCP - Protocol Reject. This local device is not configured to support bridging.

```
XMIT-3:: 8 octets @ 2C2E94
[0000]: ff 03 80 31 08 55 00 04
```

IPCP Configure Acknowledgment

```
RECV-3:: 14 octets @ 2C1634
[0000]: ff 03 80 21 02 01 00 0a 03 06 01 01 01 01
```

IPCP Configure Request - Note VJ Header Compression is not requested this time.

```
RECV-3:: 14 octets @ 2C1C4C
[0000]: ff 03 80 21 01 e4 00 0a 03 06 02 02 02 02
```

IPCP Configure Acknowledgment

```
XMIT-3:: 14 octets @ 2C2E94
[0000]: ff 03 80 21 02 e4 00 0a 03 06 01 01 01 01
```

At this point, a PPP connection has been successfully negotiated. The caller was successfully authenticated by means of CHAP and IPCP was the only successfully configured NCP. IPX, Appletalk and bridging will not be supported during this session.

Below are two packets used in determining link quality:

LCP Echo request packet

```
RECV-3:: 16 octets @ 2BEB8C
[0000]: ff 03 c0 21 09 01 00 0c 4e 36 c9 05 00 00 00 00
```

LCP Echo Response

```
XMIT-3:: 16 octets @ 2C2E94
[0000]: ff 03 c0 21 0a 01 00 0c 00 00 00 00 00 00 00 00
```

Example of MP+ call negotiation

LCP Configuration Request - MP+, MRU of 1524, MRRU of 1524, End Point Discriminator using the device's MAC address:

```
XMIT-31:: 29 octets @ D803C
[0000]: ff 03 c0 21 01 01 00 19 00 04 00 00 01 04 05 f4
[0010]: 11 04 05 f4 13 09 03 00 c0 7b 5c d3 71
```

LCP Configure Request - MP+, MRU of 1524, PAP authentication is required. MRRU of 1524, End Point Discriminator using the device's MAC address:

```
RECV-31:: 33 octets @ D4FBC
[0000]: ff 03 c0 21 01 01 00 1d 00 04 00 00 01 04 05 f4
[0010]: 03 04 c0 23 11 04 05 f4 13 09 03 00 c0 7b 53 f0
[0020]: 7a
```

LCP Configuration Acknowledgment -

```
RECV-31:: 29 octets @ D55CC
[0000]: ff 03 c0 21 02 01 00 19 00 04 00 00 01 04 05 f4
[0010]: 11 04 05 f4 13 09 03 00 c0 7b 5c d3 71
```

LCP Configuration Acknowledgment -

```
XMIT-31:: 33 octets @ D803C
[0000]: ff 03 c0 21 02 01 00 1d 00 04 00 00 01 04 05 f4
[0010]: 03 04 c0 23 11 04 05 f4 13 09 03 00 c0 7b 53 f0
[0020]: 7a
```

At this point, LCP is up. Next is the authentication phase. The local device agreed to authenticate using PAP, so it should transmit its user name and password. Note that it is not encrypted, and user name and password can be decoded very easily:

PAP Authentication Request - User name is shown in hexadecimal and must be converted to ascii. User name is 0x6a 0x73 0x6d 0x69 0x74 0x68 (jsmith) and password is 0x72 0x65 0x64 (red):

```
XMIT-31:: 20 octets @ D803C
[0000]: ff 03 c0 23 01 01 00 10 06 6a 73 6d 69 74 68 03 72
[0010]: 65 64
```

PAP Authentication Acknowledgment -

```
RECV-31:: 9 octets @ D5BDC
[0000]: ff 03 c0 23 02 01 00 05 00
```

Authentication is successful. Final negotiation determines protocols to be supported over the link.

Note: MP+ was negotiated, and both devices begin sending MP+ packets from here. The data portion of the packet is identical to PPP, but there is an 8-byte MP+ header instead of the 2-byte PPP header:

In the following packet, `00 3d` is the designation for a Multilink packet. The next byte designates whether this packet is fragmented. The next three bytes are the sequence number. You'll see them increment by one for each packet sent or received.

Next, the 80 31 01 designates this as a BCP Configure Request:

```
RECV-31:: 20 octets @ D61EC
[0000]: ff 03 00 3d c0 00 00 00 80 31 01 01 00 0a 03 03
[0010]: 01 07 03 00
```

BCP Configure Request:

```
XMIT-31:: 20 octets @ D803C
[0000]: ff 03 00 3d c0 00 00 00 80 31 01 01 00 0a 03 03
[0010]: 01 07 03 00
```

BCP Configure Acknowledgment:

```
XMIT-31:: 20 octets @ D864C
[0000]: ff 03 00 3d c0 00 00 01 80 31 02 01 00 0a 03 03
[0010]: 01 07 03 00
```

BCP Configure Acknowledgment:

```
RECV-31:: 20 octets @ D67FC
[0000]: ff 03 00 3d c0 00 00 01 80 31 02 01 00 0a 03 03
[0010]: 01 07 03 00
```

BCP is up and the session begins sending bridged traffic. No routed protocols were negotiated.

The following packets are sent as part of the MP+ protocol. They are sent at one-second intervals. These packets are used by each unit to validate the existence of the link. It gives the devices a secure way to determine whether the link is still up, even if there is no data traffic passing between the devices.

```
RECV-31:: 8 octets @ D5BDC
[0000]: ff 03 00 3d c0 00 00 05
XMIT-31:: 8 octets @ D803C
[0000]: ff 03 00 3d c0 00 00 04
RECV-31:: 8 octets @ D61EC
[0000]: ff 03 00 3d c0 00 00 06
XMIT-31:: 8 octets @ D803C
[0000]: ff 03 00 3d c0 00 00 05
```

The following RFCs provide more detail about the subjects listed in their titles:

Identifier	Title
RFC1378	PPP AppleTalk Control Protocol (ATCP)
RFC1552	PPP Internetwork Packet Exchange Control Protocol (IPXCP)

Identifier	Title
RFC1638	PPP Bridging Control Protocol (BCP)
RFC1661	Point-to-Point Protocol (PPP)
RFC1934	Ascend's Multilink Protocol Plus (MP+)
RFC1962	PPP Compression Control Protocol (CCP)
RFC1974	PPP Stac LZS Compression Protocol
RFC1989	PPP Link Quality Monitoring
RFC1990	PPP Multilink Protocol (MP)
RFC1994	PPP Challenge Handshake Authentication Protocol

FCC and International Notices

D

FCC Part 68

Ascend Communications MAX models: MAX-DSX/DSX, MAX-CSU/CSU, and MAX-CSU/DSX, have been tested to comply with Part 68 of FCC Rules. Please note the following:

1 Upon request of the telephone company, you should provide the FCC registration number of the equipment that is connected to your line. The MAX's registration number for the CSU interface(s) of the MAX-CS/DSU and MAX CSU/DSX is 2CZUSA-74422-XD-N. The MAX's registration number for the DSX interface(s) of the MAX DSX/DSX and MAX-CSU/DSX models is 2CZUSA-74421-DE-N.

2 The MAX operates with a 1.544 Mbps digital channel, using RJ48 USOC jacks. The service code is 6.0N. The Facility Interface Code is 04DU9-BN for lines using the Superframe Format (SF); 04DU9-DN for lines using the SF with B8ZS; 04DU9-1SN for lines using Extended Superframe Format (ESF) with B8ZS; and 04DU9-1KN for lines using ESF format with AMI. The MAX connects to the network using eight-pin modular plugs, wired per FCC Part 68, USOC RJ48C.

3 The telephone company must be notified before removal of a MAX connected to 1.544 Mbps digital service. If the telephone company notes a problem, they may temporarily discontinue service and will notify you of this disconnection. (If advance notice is not feasible, you will be notified as soon as possible.) When you are notified, you will be given the opportunity to correct the problem and informed of your right to file a complaint with the FCC.

FCC Part 68 Notice

This Ascend equipment complies with Part 68 of the FCC rules. Located on the equipment is a label that contains, among other information, the FCC registration number. If requested, this information must be provided to the telephone company.

This equipment cannot be used on the telephone company-provided coin service. Connection to Party Line Service is subject to State Tariffs.

If this equipment causes harm to the telephone network, the telephone company will notify you in advance that temporary discontinuance of service may be required. If advance notice isn't practical, the telephone company will notify the customer as soon as possible. Also, you will be advised of your right to file a complaint with the FCC if you believe it is necessary.

The telephone company may make changes in its facilities, equipment. operations, or procedures that could affect the operation of the equipment. If this happens, the telephone company will provide advance notice in order for you to make the necessary modifications in order to maintain uninterrupted service.

If trouble is experienced with this equipment, please contact:

Ascend Communications, Inc.
1701 Harbor Bay Parkway
Alameda, CA 94502

If the trouble is causing harm to the telephone network, the telephone company may request you to remove the equipment from the network until the problem is resolved.

It is recommended that the customer install an AC surge arrestor in the AC outlet to which this device is connected. This is to avoid damage to the equipment caused by local lightening strikes and other electrical surges.

This equipment uses the following USOC jacks and codes:

Model Name	Facility Interface Code	Service Order Code	Jack Type
TNT-SL-CT1	04DU9-BN	6.0N	RJ48C
TNT-SL-CT1	04DU9-DN	6.0N	RJ48C
TNT-SL-CT1	04DU9-1KN	6.0N	RJ48C
TNT-SL-CT1	04DU9-1SN	6.0N	RJ48C
TNT-SL-CT1	04DU9-1ZN	6.0N	RJ48C
TNT-SL-FL10	04DU9-BN	6.0N	RJ48C
TNT-SL-FL10	04DU9-DN	6.0N	RJ48C
TNT-SL-FL10	04DU9-1KN	6.0N	RJ48C
TNT-SL-FL10	04DU9-1SN	6.0N	RJ48C
TNT-SL-FL10	04DU9-1ZN	6.0N	RJ48C

FCC Part 15

Warning: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his or her own expense.

The authority to operate this equipment is conditioned by the requirement that no modifications will be made to the equipment unless the changes or modifications are expressly approved by Ascend Communications, Inc.

Canadian Notice

Note: The Canadian Department of Communications label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective, operational and safety requirements. The Department does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. In some cases, the company's inside wiring associated with a single line individual service may be extended by means of a certified connector assembly (telephone extension cord). The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situation.

Repairs to certified equipment should be made by an authorized Canadian maintenance facility designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

Caution: Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

The *Load Number (LN)* assigned to each terminal device denotes the percentage of the total load to be connected to a telephone loop which is used by the device, to prevent overloading. The termination on a loop may consist of any combination of devices subject only to the requirement that the total of the Load Numbers of all the devices does not exceed 100.

This equipment does not support line loopbacks.

Warning: THE DIGITAL APPARATUS DOES NOT EXCEED THE CLASS A LIMITS FOR RADIO NOISE EMISSIONS FROM DIGITAL APPARATUS SET OUT IN THE RADIO INTERFERENCE REGULATIONS OF THE CANADIAN DEPARTMENT OF COMMUNICATIONS.

LE PRESENT APPAREIL NUMERIQUE N'EMET PAS DE BRUITS RADIOELECTRIQUES DEPASSANT LES LIMITES APPLICABLES AUX APPAREILS NUMERIQUES DE LA CLASSE A PRESCRITES DANS LE REGLEMENT SUR LE BROUILLAGE RADIOELECTRIQUE EDICTE PAR LE MINISTERE DES COMMUNICATIONS DU CANADA.

Line Connection and Signaling - CE Notice

The MAX TNT has been approved for connection to the Public Switched Telecommunication Network using interfaces compatible with CCITT recommendations I.421 (Primary Rate ISDN user access), G.703 (DASS2 user access), and I.420 (Basic Rate ISDN user access). The MAX TNT complies with the following Council Directives:

1. Council Directive 73/23/EEC of 19 February 1973 on the harmonisation of the laws of the Member States relating to electrical equipment designed for use within certain Voltage limits. (The Low Voltage Directive)
2. The Council Directive 89/336/EEC of 3 May 1992 on the approximation of the laws of the member states relating to ElectroMagnetic Compatibility. (The EMC Directive)
3. Council Directive 91/263/EEC of 29 April 1991 on the approximation of the laws of the Member States concerning telecommunication terminal equipment. (The Telecom Terminal Equipment Directive)
4. The Council Directive 92/31/EEC of 28 April 1992 amending directive on the approximation of the laws of the member states relating to ElectroMagnetic Compatibility.
5. 93/68/EEC of 22 July 1993 amending the Directives 89/336/EEC, 91/263/EEC and 92/31/EEC. (The Marking Directive)

Manufacturer's Declaration of Conformity

Ascend Communications, Inc., hereby declares that the MAX TNT complies to the requirements of BS7378 part 1: 1991, clause 5.1.8 (No signal condition).

The following condition causes a no signal condition.

- No valid signal going from the MAX TNT to the switch.
- No signal coming into the MAX TNT from the switch.
- No power to the MAX TNT (the unit is powered down).

Warranty

E

Product warranty

1 Ascend Communications, Inc. warrants that the MAX TNT will be free from defects in material and workmanship for a period of twelve (12) months from date of shipment.

2 Ascend Communications, Inc. shall incur no liability under this warranty if

- the allegedly defective goods are not returned prepaid to Ascend Communications, Inc. within thirty (30) days of the discovery of the alleged defect and in accordance with Ascend Communications, Inc.'s repair procedures; or
- Ascend Communications, Inc.'s tests disclose that the alleged defect is not due to defects in material or workmanship.

3 Ascend Communications, Inc.'s liability shall be limited to either repair or replacement of the defective goods, at Ascend Communications, Inc.'s option.

4 Ascend Communications, Inc. MAKES NO EXPRESS OR IMPLIED WARRANTIES REGARDING THE QUALITY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE BEYOND THOSE THAT APPEAR IN THE APPLICABLE Ascend Communications, Inc. USER'S DOCUMENTATION. Ascend Communications, Inc. SHALL NOT BE RESPONSIBLE FOR CONSEQUENTIAL, INCIDENTAL, OR PUNITIVE DAMAGE, INCLUDING, BUT NOT LIMITED TO, LOSS OF PROFITS OR DAMAGES TO BUSINESS OR BUSINESS RELATIONS. THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES.

Warranty repair

1 During the first three (3) months of ownership, Ascend Communications, Inc. will repair or replace a defective product covered under warranty within twenty-four (24) hours of receipt of the product. During the fourth (4th) through twelfth (12th) months of ownership, Ascend Communications, Inc. will repair or replace a defective product covered under warranty within ten (10) days of receipt of the product. The warranty period for the replaced product shall be ninety (90) days or the remainder of the warranty period of the original unit, whichever is greater. Ascend Communications, Inc. will ship surface freight. Expedited freight is at customer's expense.

2 The customer must return the defective product to Ascend Communications, Inc. within fourteen (14) days after the request for replacement. If the defective product is not returned within this time period, Ascend Communications, Inc. will bill the customer for the product at list price.

Out-of warranty repair

Ascend Communications, Inc. will either repair or, at its option, replace a defective product not covered under warranty within ten (10) working days of its receipt. Repair charges are

available from the Repair Facility upon request. The warranty on a serviced product is thirty (30) days measured from date of service. Out-of-warranty repair charges are based upon the prices in effect at the time of return.

Index

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

X