Pipeline 220 VT100 Interface Configuration Guide

Ascend Communications, Inc.
Part Number: 7820-0324-001
For Software Version 5.1Ap5

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
- Whether you are routing or bridging with your Ascend product
- 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`

Contents

Figures

Tables

About This Guide

This guide describes how to configure the Pipeline 220 using the VT100 interface.

What you should know

This guide is for the person who configures and maintains the Pipeline. To configure the Pipeline, you need to understand the following:

- Internet or telecommuting concepts
- Wide area network (WAN) concepts
- Local area network (LAN) concepts

Documentation conventions

This section explains all the special characters and typographical conventions in this manual.

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 name of an item you select from a menu. This symbol appears between the name of a menu and the name of the item you should select from the menu. (The *menu* does not necessarily appear at the top of the screen. For example, you might open it by clicking a button.)

Convention	Meaning
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.

Manual set

The Pipeline 220 Documentation Set consists of the following manuals:

- *Pipeline 220 User's Guide*
  Explains how to install the hardware and configure the Pipeline using the Java Configurator.
- *Pipeline 220 VT100 Interface Configuration Guide*
  Explains how to use the VT100 user interface to configure the Pipeline 220.
- *Pipeline 220 VT100 Interface Reference Guide*
  Describes each parameter and command in the VT100 user interface.

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 2153: *PPP Vendor Extensions*
- RFC 1994: *PPP Challenge Handshake Authentication Protocol (CHAP)*
- RFC 1990: *The PPP Multilink Protocol (MP)*
- RFC 1989: *PPP Link Quality Monitoring*
- RFC 1974: *PPP Stac LZS Compression Protocol*
- RFC 1962: *The PPP Compression Control Protocol (CCP)*
- RFC 1877: *PPP Internet Protocol Control Protocol Extensions for Name Server Addresses*

- RFC 1662: *PPP in HDLC-like Framing*
- RFC 1661: *The Point-to-Point Protocol (PPP)*
- RFC 1638: *PPP Bridging Control Protocol (BCP)*
- RFC 1332: *The PPP Internet Protocol Control Protocol (IPCP)*

Information about IP routers

RFCs that describe the operation of IP routers include:

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

Information about OSPF routing

For information about OSPF routing, see:

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

Information about multicast

For information about multicast, see:

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

Information about packet filtering

RFCs that describe firewalls and packet filters include:

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

Information about general network security

RFCs pertinent to network security include:

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

ITU-T recommendations

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

Related publications

This guide and documentation set do not provide a detailed explanation of products, architectures, or standards developed by other companies or organizations.

Here are some related publications that you might find useful:

- William Flanagan. *The guide to T1 Networking*.
- Uyless Black. *Data Link Protocols*
- W. Richard Stevens. *TCP/IP Illustrated*
- William R. Cheswick and Steven M. Bellovin. *Firewalls and Internet Security*

Introduction

1

This chapter covers the following topics:

Using the Pipeline for private and public access

The Pipeline is a high-performance WAN router that can enable all Internet users to access your FTP site, World Wide Web site, and any other publicly available resources, while your employees have secure access to your corporate network backbone.

The Pipeline delivers WAN access through either an unchannelized T1/FT1 or a V.35 interface. You cannot use both WAN interfaces simultaneously. Your software configuration activates one or the other.

The most common users of the Pipeline are medium to large companies, major corporations, and ISPs who provide both open access and secure access through multiple Ethernet LAN segments. The unit's configuration options provide the flexibility and security you need to optimize your installation. Management features include a comprehensive set of control and monitoring functions and easy-to-perform upgrades.

Common applications for the Pipeline

Figure 1-1 through Figure 1-4 show a variety of applications using the features of the Pipeline.

Dual LAN access

Figure 1-1shows a typical configuration for a company that offers publicly accessible network resources to all employees and restricted access to a separate secure network

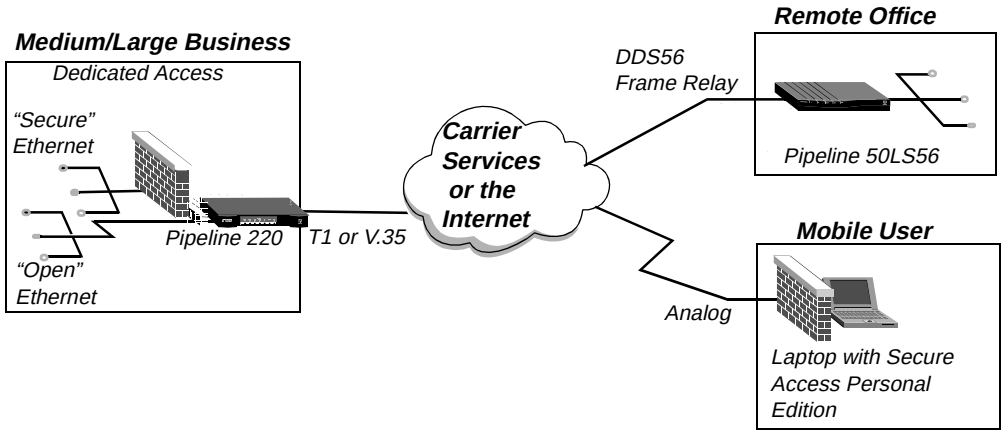


Figure 1-1. Dual LAN access for employees in a corporation

The figure shows two types of users: a remote office that connects to the corporate network through a leased Frame Relay connection, and a remote user who dials into the Internet through an ISP connection and connects to the corporate network by means of the Internet.

WWW access for all Internet users

Figure 1-2 shows a typical configuration for a company that offers a World Wide Web site and FTP site to any Internet user and a secure corporate network connection to its employees

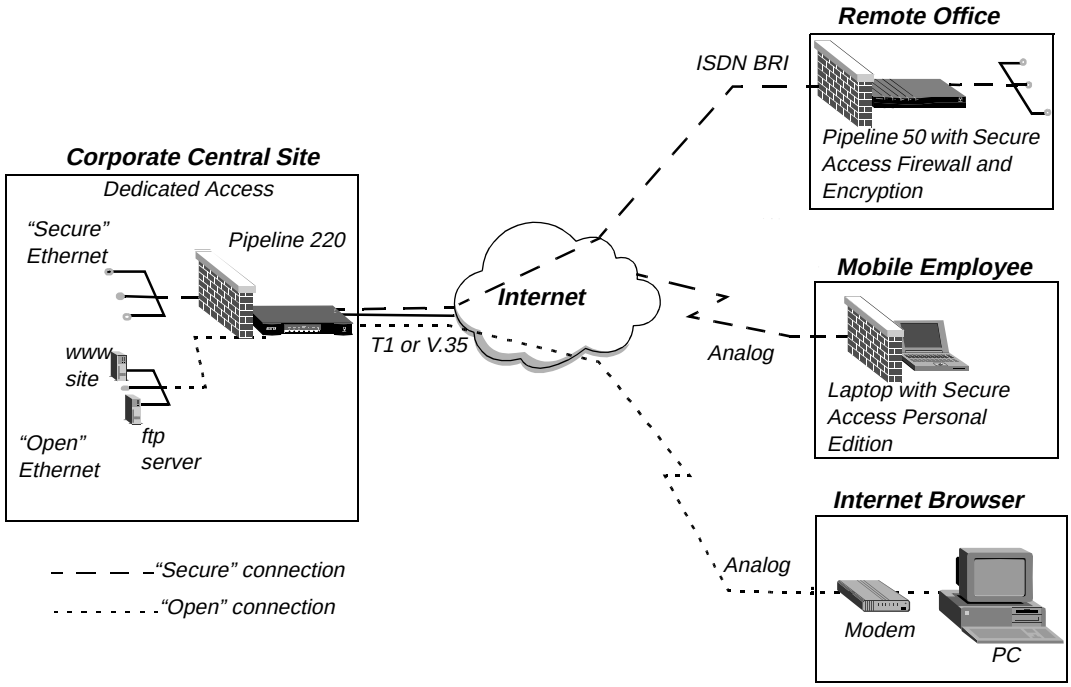


Figure 1-2. Public access for Internet users and private access for employees

You can ensure security by using built-in static filters, Connection profiles, and RADIUS, or you can use the optional Secure Access feature to ensure the highest level of security through Ascend's Secure Access Firewalls and IPSec encryption.

Virtual Private Networking (VPN)

With the optional VPN feature, you can forward non-IP traffic across the Internet. The feature supports the following protocols:

- Point To Point Tunneling Protocol (PPTP)

  Note: The Pipeline supports PPTP by routing or forwarding PPTP traffic as appropriate. The Pipeline does not act as either a PPTP Access Concentrator (PAC) or a PPTP Network Server (PNS).

- Ascend Tunnel Management Protocol (ATMP)

Figure 1-3 shows a typical network tunneling environment.

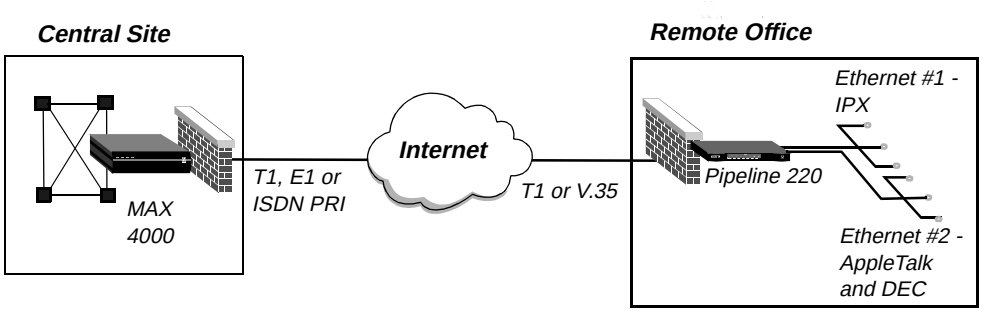


Figure 1-3. Network tunneling across the Internet

Network tunneling adds another level of security by encrypting the data it sends across the Internet. The supported tunneling protocols can be used in combination with Secure Access.

Internet Gateway

Figure 1-4 shows a company offering switched access to corporate resources for remote offices, telecommuters, and mobile users through an Ascend MAX. The company offers its employees protected connection to the Internet through the Pipeline:

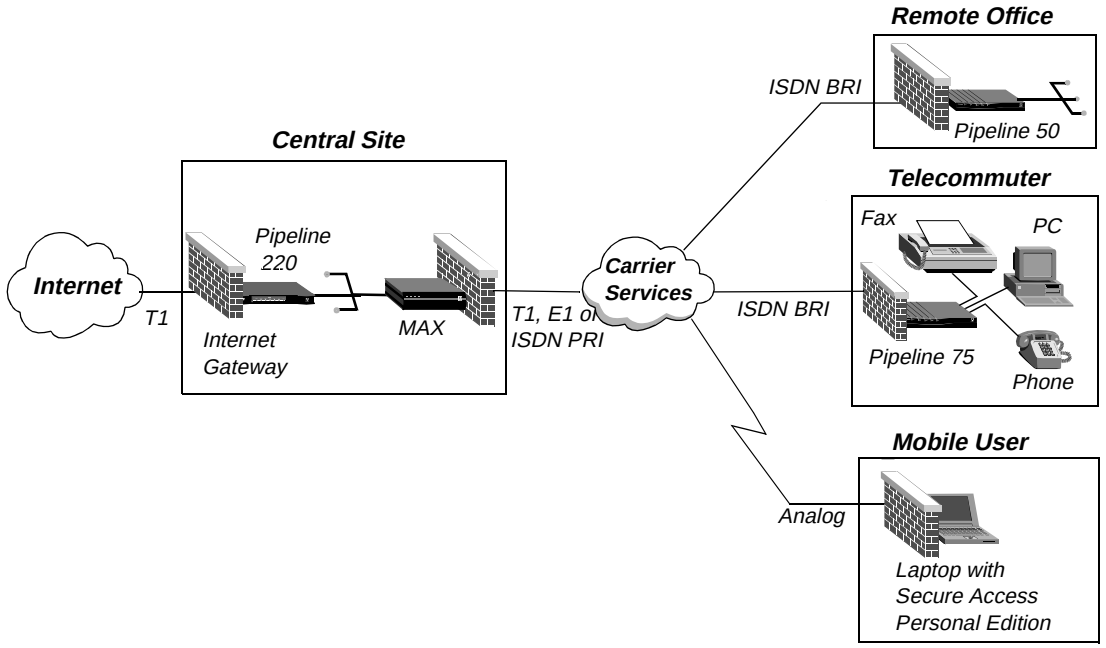


Figure 1-4. Using the Pipeline as a central-site Internet gateway

Overview of Pipeline configuration

This section provides an overview of how to configure the Pipeline. It covers the following topics:

- Configuring the lines, channels, and ports, and how calls are routed between them
- Configuring wide area network connections and security
- Configuring the Pipeline as a Frame Relay or X.25 concentrator
- Configuring routing and bridging across the WAN
- Configuring Internet services, such as multicast, OSPF, and virtual private networks

Creating a network diagram

Ascend strongly recommends that, after you have read this introductory material, you diagram your network and refer to the diagram while configuring the Pipeline.

Creating a comprehensive network diagram helps prevent problems during installation and configuration, and can help you troubleshoot problems later.

Configuring lines, slots, and ports for WAN access

The Pipeline comes with one built-in T1 or E1 line and a V.35 serial port (8 Mbps). You cannot simultaneously use both types of access.

You can use either type of access for a leased high-speed connection to a Frame Relay switch or to another WAN router. Neither type requires extensive configuration. Your service provider will provide you with the small amount of information you need to configure the Pipeline for WAN access. You specify most of the required information in a Frame Relay or Connection profile.

Once you have enabled the lines and ports for WAN access, you need to configure the manner in which users are routed to them, for access across the WAN, and routed from them to other destinations (such as the local network).

Configuring WAN connections and security

When the Pipeline receives packets that require routing to a remote network, it forwards them across the WAN connection. Software at the both ends of the connection encapsulates each packet before sending it out over the WAN. Each type of encapsulation supports its own set of options, which can be configured on a per-connection basis to enable the Pipeline to interact with a wide range of software and devices.

After a connection's link encapsulation method has been negotiated, the Pipeline typically uses a password to authenticate the call. Following are some of the connection security features supported in the Pipeline:

Feature	Description
Authentication protocols	For PPP connections, the Pipeline supports both Password Authentication Protocol (PAP) and Challenge-Handshake Authentication Protocol (CHAP). CHAP is more secure than PAP, and is preferred if both sides of the connection support it.
Terminal server security	After a dial-in user has passed the initial-connection security check, another password can be required for access to the Pipeline terminal services. Within the terminal server, you can restrict which commands are accessible to users, or prevent users from executing any command other than Telnet.
Filters and firewalls	Filters and firewalls provide a packet-level security mechanism that can provide a very high level of network security.

Concentrating Frame Relay connections

The Pipeline provides extensive support for Frame Relay. Using a T1 line or serial WAN port for a nailed connection to a switch, it can function as an Network-to-Network Interface (NNI) switch, a Data Communications Equipment (DCE) unit responding to users, or as a Data Terminal Equipment (DTE) requesting services from a switch.

Configuring routing and bridging across the WAN

Routing and bridging configurations enable the Pipeline to forward packets between the local network and the WAN.

Protocol-independent packet bridging

The Pipeline can operate as a link-level bridge, forwarding packets from Ethernet to a WAN connection (and vice versa) on the basis of the destination hardware address in each packet. Unlike a router, a bridge does not examine packets at the network layer. It simply forwards packets to another network segment if the address does not reside on the local segment.

IPX routing (NetWare 3.11 or newer)

The Pipeline can operate as an IPX router, linking remote NetWare LANs with the local NetWare LAN on Ethernet.

IP routing

IP routing is the most widespread use of the Pipeline, and the unit has a wide variety of configurable options. IP routing is the required basis for Internet-related services such as IP multicast support, OSPF, and cross-Internet tunneling for virtual private networks. Most sites create static IP routes to enable the Pipeline to reliably connect to certain destinations or to change global metrics or preferences settings.

Configuring Internet services

All Internet services and routing methods require that the Pipeline function as an IP router, so an IP routing configuration is a necessary precondition.

Multicast

The multicast backbone (MBONE) is a virtual network layered on top of the Internet to support IP multicast routing across point-to-point links. It is often used for transmitting audio and video on the Internet in real-time, because multicasting is a much cheaper and faster way to communicate the same information to multiple hosts.

OSPF routing

Open Shortest Path First (OSPF) is the next generation Internet routing protocol. You can configure the Pipeline to communicate with other OSPF routers within an autonomous system (AS). To enable this routing function, you must configure the OSPF options on the Ethernet interface and for each WAN connection that supports remote OSPF routers.

OSPF can import routes from RIP as well. You can control the way these imported external routes are handled by adjusting system-wide routing options such as route preferences and ASE-type metrics.

Virtual Private Networking (VPN)

Many sites use the Internet to connect corporate sites or to enable mobile nodes to log into a corporate backbone. Such virtual private networks use cross-Internet tunneling to maintain security or to enable the Internet to transport protocols that it would otherwise drop, such as IPX. To implement virtual private networks, the Pipeline, with the VPN option, supports the Ascend Tunneling Management Protocol (ATMP) and the Point-to-Point Tunneling Protocol (PPTP).

ATMP enables the Pipeline to create and tear down a tunnel to another Ascend unit. In effect, the tunnel collapses the Internet cloud and provides what looks like direct access to a home network. Packets received through the tunnel must be routed, so ATMP applies only to IP or IPX networks at this time.

Note: The Pipeline supports PPTP by routing or forwarding PPTP traffic as appropriate. The Pipeline does not act as either a PPTP Access Concentrator (PAC) or a PPTP Network Server (PNS).

Overview of management features

This section describes the following management functions, which use features built into the Pipeline.

- Using the Ascend Configurator
- Using the terminal server command line
- Using status windows to track WAN or Ethernet activity
- Managing the Pipeline by means of SNMP
- Using remote management to configure far-end Ascend units
- Updating software in the Pipeline unit's flash RAM

The Pipeline provides up to nine security levels to control which management and configuration functions users can access.

Using the Ascend Configurator

To configure the Pipeline, you use the Ascend Configurator. This application is easily installed on your Windows NT workstation or Windows 95 workstation. The configurator enables you to:

- Configure the Pipeline for the first time.
- Modify a pre-configured Pipeline.
- Save any Pipeline configuration to a text file.

See the *Pipeline 220 User's Guide* for more details.

Using the terminal server command line

To invoke the terminal server command-line interface, you must use the VT100 interface and must have administrative privileges. Once you have activated a Security profile that enables the necessary privileges, you can invoke the command line by selecting Term Serv in the Sys Diag menu. To close the command-line, enter the Quit command at the command-line prompt. The cursor then returns to the VT100 menu interface.

Using status windows to track WAN or Ethernet activity

In the Pipeline configuration menus, the right side of the screen displays eight status windows. The windows provide a great deal of read-only information about what is currently happening

in the Pipeline. If you want to focus on the activity of a particular slot card, you can change the default contents of the windows to show what is currently going on in that slot.

See the *Pipeline 220 VT100 Interface Reference Guide* for more details.

Managing the Pipeline by means of SNMP

Many sites use Simple Network Management Protocol (SNMP) applications to obtain information about the Pipeline, and use the information to enhance security, set alarms for certain conditions, and perform simple configuration tasks.

The Pipeline supports the Ascend Enterprise MIB, MIB II, and some ancillary SNMP features. The Pipeline can send management information to an SNMP manager without being polled. SNMP security uses a community name sent with each request. The Pipeline supports two community names, one with read-only access, and the other with read/write access to the MIB.

Flash RAM and software updates

Flash RAM technology enables you to perform software upgrades in the field without opening the unit or changing memory chips. You can upgrade the Pipeline through its serial control port, or through its Ethernet interface use Trivial File Transfer Protocol (TFTP) to upgrade the unit.

Where to go next

When you have planned your network, you are ready to configure the Pipeline. The flexibility of the Pipeline and its ever-increasing number of configurations options means there is no set order for configuration. You can perform configuration tasks in any order you want. Figure 1-5 shows you where to look for the information you need:

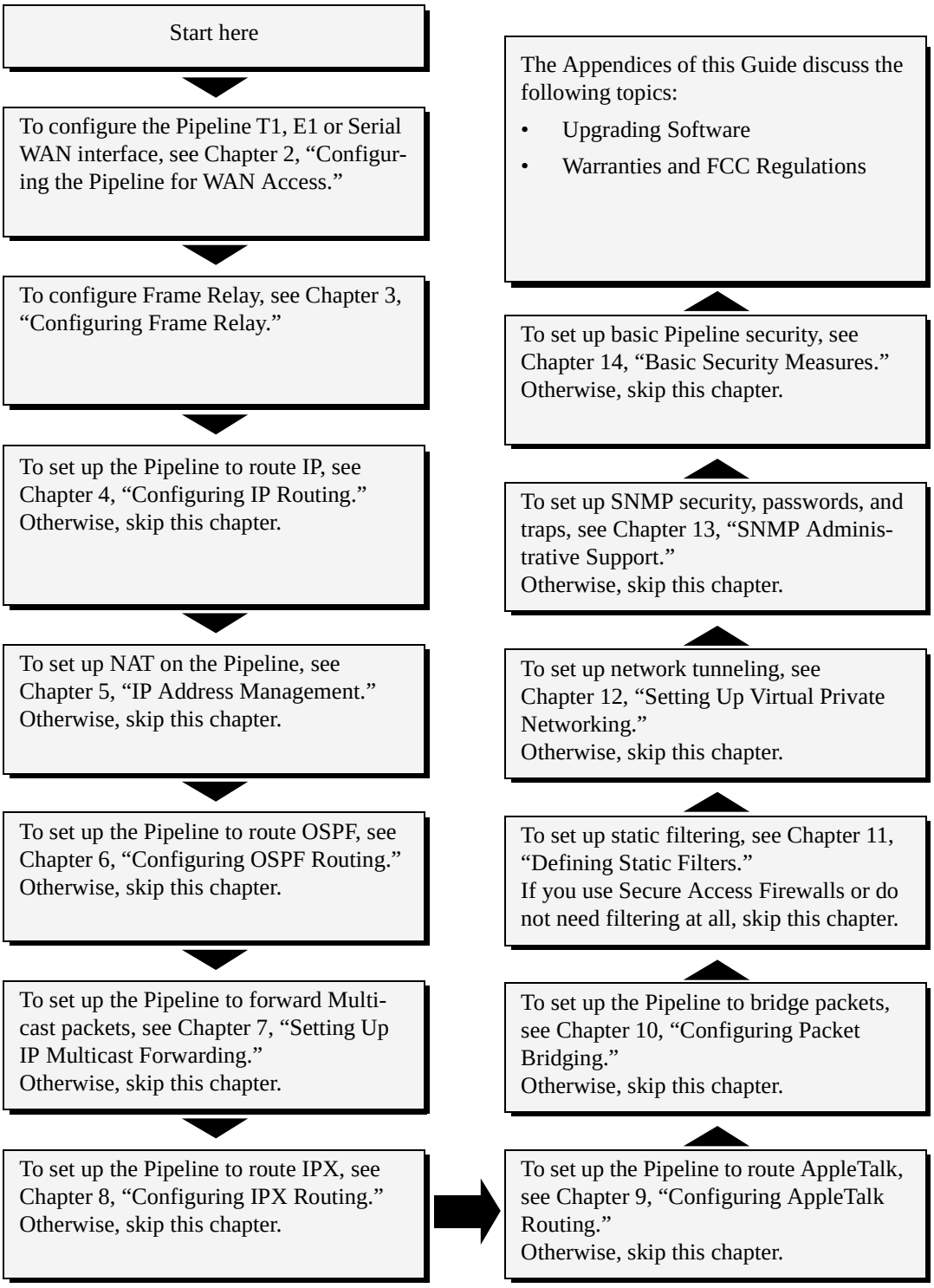


Figure 1-5. Pipeline VT100 Interface Configuration Guide Roadmap

Configuring the Pipeline for WAN Access 2

This chapter covers these topics:

Introduction to WAN configuration

The Pipeline comes with a built-in T1 or E1 connection and a V.35 serial port. You must configure one or the other for WAN access.

Figure 2-1 shows the Pipeline back panel. The T1 or E1 connection is the RJ-45 port labelled WAN1. The V.35 serial port is labelled WAN2.

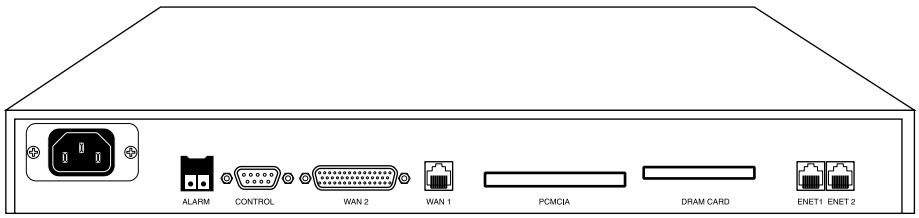


Figure 2-1. Back panel of the Pipeline

Specifying a WAN interface

To specify a WAN interface:

1 From the Main Edit window, select System > Sys Config:

2 Set WAN Interface to appropriate interface:

- T1-CSU specifies nailed T1 interface
- E1-Nailed specifies nailed E1 interface
- Serial WAN specifies a Serial WAN interface

3 Exit and save the Sys Config profile.

4 From the System menu, select Sys Diag.

5 Select Sys Reset and press Enter.

After the Pipeline resets, the selected interface is enabled. Next, you must configure the WAN interface:

To configure the T1-CSU port, see "Configuring the T1 line," next.

To configure the E1 port, see "Configuring the E1 line" on page 2-4.

To configure the Serial WAN port, see "Configuring the serial WAN port" on page 2-6.

Configuring the T1 line

The T1 connection (WAN1) Pipeline is not channelized, but you can configure it like a T1 with any number of DS0 channels, up to 24, as specified by your carrier.

This section describes how to configure the Serial Port T1-CSU (Channel Service Unit) line in the Pipeline. If the unit is configured for serial WAN instead, skip this section (see "Configuring the serial WAN port" on page 2-6).

With a nailed T1 line, you must manually configure some port information. For example, you must specify the signals that indicate that the Data Communications Equipment (DCE) is ready to connect. In addition, you might need to adjust the amount of attenuation that the Pipeline should apply to the line's network interface in order to match the cable length from the Pipeline to the next repeater.

To configure the nailed T1 line, you perform the following tasks:

- Supply information, such as encoding, framing, and buildout (attenuation) that you obtain from your carrier.
- Activate the port.

For complete information about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.

This section provides background information about the T1 line interface parameters.

T1 line framing and encoding

The framing used by the physical layer of the T1 line may be D4 or ESF. D4 format, also known as the superframe format, consists of 12 consecutive frames separated by framing bits. The line must not use ISDN signaling with D4 framing, because false framing and Yellow Alarm emulation can result. ESF specifies the extended superframe format. This format consists of 24 consecutive frames separated by framing bits. The ISDN specification advises that you use ESF with ISDN D-channel signaling.

The encoding value sets the layer-1 line encoding used for the physical links, which affects the way data is represented by the digital signals on the line. Your carrier can tell you which encoding to use. AMI (the default) specifies Alternate Mark Inversion encoding. B8ZS specifies that the encoding is Bipolar with 8-Zero Substitution. The other option, None, is identical to AMI but without density enforcement.

Amount of attenuation required

The Buildout parameter specifies the amount of attenuation to apply to the T1 transceiver's internal CSU. The amount depends on the cable length from the Pipeline to the next repeater. Valid values are 0 db (decibels) through 22.5 db.

Attenuation is a measure of the power lost on a transmission line or on a portion of that line. When you specify a value for Buildout, the Pipeline applies attenuation to the T1 line, causing the line to lose power. Repeaters boost the signal on a T1 line. If the Pipeline is too close to a repeater, you might need to add some attenuation. Check with your carrier to determine the correct value.

Clock source for synchronous transmission

The Clock Source parameter indicates whether the T1 line can be used as the master clock source for synchronous connections. In synchronous transmission, both the sending device and the receiving device must maintain synchronization in order to determine where one block of data ends and the next begins.

You might need to disable this parameter on one unit if two Ascend units connect to each other by a crossover cable (with optional T1 repeaters) between their network ports.

Number of channels in the line

Specifies the number of channels provisioned for your line. Check with your carrier to determine the correct value.

Configuring the nailed T1 line

To configure the nailed T1 line, proceed as in the following example:

1 Open the Serial Port T1-CSU Profile.

2 Open the Mod Config submenu.

3
```
Nailed-T1 Group=1
Activation=Enabled
Framing Mode=D4
Encoding=B8ZS
FDL=None
Build Out=0db
Clock Source=Yes
Number of DS0 Channels=6
```

4 Leave the Nailed T1 Group set to 1.

```
Nailed T1 Group=1
```

Note: The Pipeline only supports a value of 1 for this parameter. A Connection profile uses this permanent link by specifying the nailed channels' group number in the Group parameter. A Frame Relay profile uses a permanent nailed link by specifying the group number in its Nailed Group parameter.

5 Activate the T1 line.

```
Activation=Enabled
```

6 Set the T1 framing mode.

```
Framing Mode=D4
```

7 Set the Encoding parameter as specified by your carrier.

```
Encoding=B8ZS
```

8 Specify the FDL used for this line.

9 Set the buildout if appropriate.

```
Build Out=0db
```

10 Specify the Clock Source.

```
Clock Source=Yes
```

11 Enter the number of DS0 channels assigned to this line by your carrier.

```
Number of DS0 Channels=24
```

12 Exit and save the Serial Port T1-CSU profile.

Using T1 line diagnostics

The Pipeline provides the following T1 status windows to diagnose the connection:

```
10-000 Ser T1-CSU
   10-100 Line 1 Stat
   10-200 Line Error
   10-300 FDL1 Stats
```

You can use these commands to gather information about the line. They are located in the Serial T1-CSU status menu. For details about each option, see the *Pipeline 220 VT100 Interface Reference Guide*.

Configuring the E1 line

The E1 connection is the RJ-45 port on the back panel of the Pipeline unit that is labelled WAN 1. This connection is not channelized, but you can configure it to act like E1 with any number of DS0 channels, up to 32, as specified by your carrier.

This section describes how to configure the Serial Port E1-Nailed line in the Pipeline. (If the unit is configured for serial WAN instead, skip this section and see “Configuring the serial WAN port” on page 2-6).

With a nailed E1 line, you must manually configure some port information; for example, you must specify the signals that indicate that the DCE is ready to connect. In addition, you might need to indicate the cable length from the Pipeline to the CSU.

To configure the nailed E1 line, you perform the following tasks:

- Specify a group number associated with the nailed E1 line
  You assign a group number to the line and then specify that group number in Connection Profiles that will access the WAN across this interface.
- Supply carrier information, such as encoding, framing, and buildout (attenuation).
- Activate the port.

For details on each parameter discussed in the following section, see the *Pipeline 220 VT100 Interface Reference Guide*.

E1 framing

The framing used by the physical layer of the E1 line may be G.703, which is the standard framing mode used by most E1 ISDN providers and by DASS 2, or 2DS, a variant of G.703 required by most E1 DPNSS providers in the U.K.

Clock source for synchronous transmission

This determines whether the E1 line can be used as the master clock source for synchronous connections. In synchronous transmission, both the sending device and the receiving device must maintain synchronization in order to determine where one block of data ends and the next begins.

How the DS0s are used

You must specify how the DSOs are used. Ending DS0 Channel specifies the last channel in your line. Enable Channel 16 specifies whether channel 16 is used for data, or whether the Pipeline should ignore it.

Configuring the nailed E1 line

To configure the Serial Port E1-Nailed line, proceed as in the following example:

1 Open the Serial Port E1-Nailed Profile.

2 Open the Mod Config submenu.

```
Mod Config...
   Nailed-E1 Group=1
   Activation=Enabled
   Framing Mode=G.703
   Clock Source=Yes
   Ending DS0 Channel=31
   Enable Channel 16=Yes
```

3 Leave Nailed-E1 Group to set to 1.

```
Nailed-E1 Group=1
```

Note: The Pipeline only supports a value of 1 for this parameter. A Connection profile uses this permanent link by specifying the nailed channels' group number in the Group parameter. A Frame Relay profile uses a permanent nailed link by specifying the group number in its Nailed Group parameter.

4 Activate the E1 line.

```
Activation=Enabled
```

5 Specify the Clock Source.

```
Clock Source=Yes
```

6 Set the E1 framing mode.

```
Framing Mode=G.703
```

7 Enter the number of DS0 channels assigned to this line by your carrier.

```
Number of DS0 Channels=24
```

8 Specify whether channel 16 is enabled on the E1 line:

```
Enable Channel 16=Yes
```

9 Exit and save the Serial Port E1-Nailed profile.

10 Set Activation to Enabled.

11 Set the Framing Mode as specified by your carrier.

12 Select Clock Source if this synchronous timing should be used from this line.

In most cases, you should set Clock Source to Yes. Specifying No indicates to the Pipeline that it should generate timing using its internal clock. Only in some back-to-back configurations should you configure the Pipeline to generate timing.

13 Enter the number of the highest DS0 channel assigned to this line by your carrier.

Any value of from 1 to 32 is valid.

Using E1 line diagnostics

The Pipeline provides the following E1 status windows to diagnose the connection:

```
10-000 Ser E1-Nailed
   10-100 Line 1 Stat
   10-200 Line Error
   10-300 Net Options
```

You can use these commands to gather information about the line. They are located in the Serial E1-Nailed status menu. For details about each option, see the *Pipeline 220 VT100 Interface Reference Guide*.

Configuring the serial WAN port

The Pipeline has a built-in V.35 serial WAN DB-44 port. A serial WAN port provides a V.35/ RS-449 WAN interface that is typically used to connect to a Frame Relay switch. The serial WAN data rate is determined by the clock speed received from the link. The maximum acceptable clock is 8 Mbit/s.

Assigning a group number to the serial WAN bandwidth

The Nailed Grp parameter assigns a number that can be referenced as the Group in a Connection profile or the Nailed Grp in a Frame Relay profile. If it is specified in a Connection profile, the Pipeline will bridge or route packets to another unit across that nailed connection. If it is used in a Frame Relay profile, the Pipeline will have a nailed connection to a frame relay switch and the DLCI number in each frame will determine which frames are sent over the link.

The number you assign must be unique in the Pipeline configuration. Do not use a group number that is already in use for a nailed connection on another interface.

Signals to control the serial WAN data flow

The Activation parameter tells the Pipeline which signals control the data flow through the serial WAN port. The DCE to which the serial WAN port is connected (such as a Frame Relay switch) determines how to set the value. Flow control always uses handled by the Clear To Send (CTS) signal.

For details about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.

Configuring the serial WAN interface

To configure the serial WAN interface to connect to a frame relay switch that uses Static data flow, proceed as in the following example:

1. Open Serial WAN>Mod Config.
2. Assign an (optional) module name and a group number.

   ```
   Module Name=wan-serial
   ```

3. Leave Nailed Grp to set to 1.

   ```
   Nailed Grp=1
   ```

   Note: The Pipeline only supports a value of 1 for this parameter. A Connection profile uses this permanent link by specifying the nailed channels' group number in the Group parameter. A Frame Relay profile uses a permanent nailed link by specifying the group number in its Nailed Group parameter.

4. Set the Activation parameter to Static.

   ```
   Activation=Static
   ```

5. Close the Serial WAN profile.
6. Configure a Frame Relay profile and specify the Nailed Grp number assigned to this port. For example:

   ```
   Frame Relay
      Name=NNI
      Active=Yes
      Call Type=Nailed
      FR Type=NNI
      LinkUp=Yes
      Nailed Grp=1
      ...
   ```

For a complete explanation of configuring a Frame Relay profile, see Chapter 3, "Configuring Frame Relay."

Configuring Frame Relay

3

This chapter contains these topics:

Using the Pipeline as a Frame Relay concentrator

In a Frame Relay backbone, every access line connects directly to a Frame Relay switch. In the past, most connections to the Frame Relay network were relatively high speed, such as full T1 or E1 lines. But with recent changes in Frame Relay pricing, many sites now want to concentrate many low-speed dial-in connections into one high-speed nailed connection to a Frame Relay switch. When the Pipeline is configured as a Frame Relay concentrator, it accepts incoming dial-in connections as usual, and forwards them out to a Frame Relay switch.

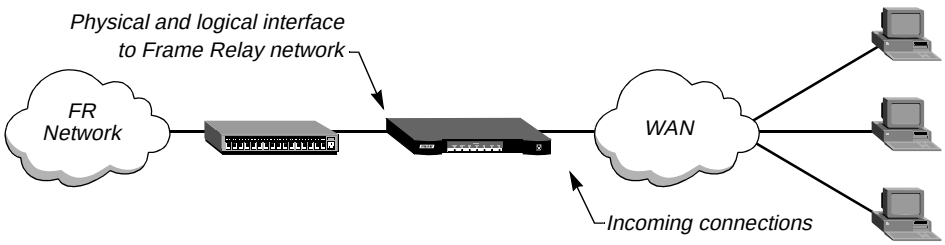


Figure 3-1. The Pipeline operating as a Frame Relay concentrator

As a Frame Relay concentrator, the Pipeline can accept many low-speed connections and concentrate them onto the single 2-Mbps serial WAN interface.

Configuring the Pipeline as a Frame Relay concentrator involves the following elements:

- An interface to the Frame Relay switch (usually nailed T1, nailed E1, or serial WAN)
- A logical datalink to the Frame Relay switch (defined in a Frame Relay profile)
- User connections (defined in Connection profiles)

For information about monitoring and managing Frame Relay, see the system adminstration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

Kinds of physical network interfaces

The Pipeline typically uses serial WAN, nailed T1, or nailed E1 to connect to a Frame Relay switch. For the details of configuring these interfaces, see Chapter 4, “Configuring the Pipeline 220 for WAN Access.”

Kinds of logical interfaces to a Frame Relay switch

The Pipeline supports NNI, UNI-DCE, and UNI-DTE interfaces to the Frame Relay network.

Network to Network Interface (NNI)

With an NNI connection allows the Pipeline appears to the switch to be a Frame Relay network interface. It performs both DTE and DCE link management, and allows two separate Frame Relay networks to connect via a common protocol. (To configure the interface, see “Configuring an NNI interface” on page 3-5.)

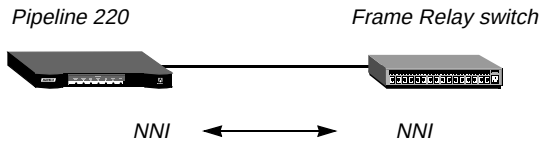


Figure 3-2. Network to Network interface (NNI) in a Pipeline unit

User to Network Interface — Data Communications Equipment (UNI-DCE)

UNI is the interface between an end-user and a network end point (a router or a switch) on the Frame Relay network. In a UNI-DCE connection, the Pipeline operates as a Frame Relay router communicating with a DTE device. To the DTE devices, it appears as a Frame Relay network end point. (To configure the interface, see “Configuring a UNI-DCE interface” on page 3-6.)

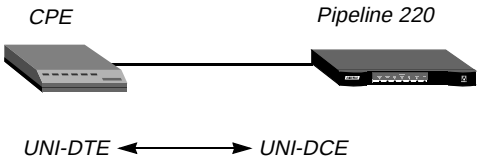


Figure 3-3. User to Network Interface-Data Communications Equipment (UNI-DCE)

User to Network Interface — Data Terminal Equipment (UNI-DTE)

In a UNI-DTE connection, the Pipeline is configured as a UNI-DTE communicating with a Frame Relay switch. It acts as a Frame Relay feeder, and performs the DTE functions specified

for link management. To configure the interface, see “Configuring a UNI-DTE interface” on page 3-6.)

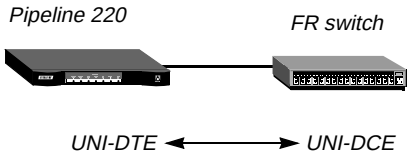


Figure 3-4. User to Network Interface - Data Terminal Equipment (UNI-DTE)

Types of Frame Relay connections

For Frame Relay connections, the Pipeline supports gateway connections and Frame Relay circuits.

Gateway connections

With a gateway connection, the Pipeline receives an incoming PPP call, examines the destination IP address, and brings up the appropriate Connection profile for that destination, as usual. If the Connection profile specifies Frame Relay encapsulation, the Frame Relay profile, and a DLCI, the Pipeline encapsulates the packets in Frame Relay (RFC 1490), placing the DLCI in the headers, and forwards the data stream out to the Frame Relay switch. The Frame Relay switch uses the DLCI to route the frames. This is known as gateway mode.

Frame Relay circuits

A Frame Relay circuit is a permanent virtual circuit (PVC) segment that has two DLCI end points and a single Frame Relay profile. It requires two and only two DLCI numbers. Data is dropped if the circuit has only one DLCI. If more than two are defined, only two are used. A circuit is defined in two Connection profiles. Data coming in on the DLCI configured in the first Connection profile is switched to the DLCI configured in the second one.

Configuring the logical link to a Frame Relay switch

The Frame Relay profile specifies a link, usually across a single cable, to the Frame Relay network. This link can support many permanent virtual circuits (PVCs), each with a different endpoint.

Understanding the Frame Relay parameters

This section provides some background information about configuring the logical link to a Frame Relay switch. (For more detailed descriptions of the parameters, see the *Pipeline 220 VT100 Interface Reference Guide*.)

Specifying a Frame Relay profile name and activating the profile

Connection profiles link with the Frame Relay profiles on the basis of Frame Relay profile name. The name must be unique and cannot exceed 15 characters.

Bringing down the datalink when DLCIs are not active

The Link Up parameter indicates that the datalink comes up automatically and stays up even when the last DLCI has been removed. If this parameter is set to No, the datalink does not come up unless a Connection profile (DLCI) brings it up, and it shuts down after the last DLCI has been removed.

Note: You can start and drop Frame Relay datalink connections by using the DO DIAL and DO HANGUP commands from the VT100 interface. DO DIAL brings up a datalink connection. DO HANGUP closes the link and any DLCIs on it. If LinkUp=Yes, DO HANGUP brings the link down, but it will be automatically restarted. A restart will also occur if there is a DLCI profile invoking the datalink. For more information, see the system administration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

Defining the nailed connection to the switch

Nailed is the default value for Frame Relay connections. When the call type is nailed, dial numbers and other telco options are N/A. You can specify switched if the Frame Relay switch allows dial-in; however, Frame Relay networks currently have no dial-out connection capability. The two types of data service that are available are 64K or 56K.

Specifying the type of Frame Relay interface

You can set the FR Type parameter to NNI (for an NNI interface to the switch), DCE (for a UNI-DCE interface), or DTE (for a UNI-DTE interface). See “Kinds of logical interfaces to a Frame Relay switch” on page 3-2.

Link management protocol

The Link Mgmt setting may be None (no link management), T1.617D (for T1.617 Annex D), and Q.933A (for Q.933 Annex A).

Frame Relay timers and event counts

Frame Relay timers and event counts are located by clicking the Frame Relay button, then the Link Management button, as follows:

- N391 specifies the interval at which the Pipeline requests a Full Status Report (between 1 and 255 seconds). It is N/A if FR Type is DCE.
- DCE N392 specifies the number of errors during DCE N393 monitored events which causes the network side to declare the user side procedures inactive. Its value should be less than DCE N393 (between 1 and 10). It is N/A when FR Type is DTE.
- DCE N393 specifies the DCE monitored event count (between 1 and 10). It is N/A when FR Type is DTE.
- DTE N392 specifies the number of errors during DTE N393 monitored events which cause the user side to declare the network side procedures inactive. Its value should be less than DTE N393 (between 1 and 10). It is N/A when FR Type is DCE.
- DTE N393 specifies the DTE monitored event count (between 1 and 10). It is N/A when FR Type is DCE.
- T391 specifies the Link Integrity Verification polling timer (between 5 and 30 seconds). Its value should be less than T392. It is N/A when FR Type is DCE.

- T392 specifies the time for Status Enquiry messages (between 5 and 30 seconds). An error is recorded if no Status Enquiry is received within T392 seconds. This parameter is N/A when FR Type is DTE.

MRU (Maximum Receive Units)

The MRU parameter specifies the maximum number of bytes the Pipeline can receive in a single packet across this link. Usually the default 1532 is the correct setting, unless the far end device requires a lower number.

Example Frame Relay profile configurations

This section shows an example Frame Relay profile configuration for each type of Frame Relay interface (NNI, UNI-DCE, and UNI-DTE).

Configuring an NNI interface

In the following example, the Pipeline has a nailed connection to another Frame Relay switch and will be configured with an NNI interface to that switch. The sample network looks like this:

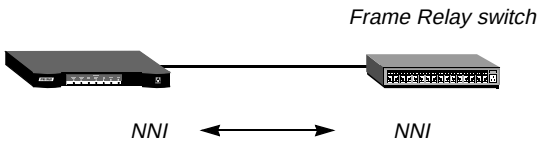


Figure 3-5. Example NNI interface to another switch

To configure the Frame Relay profile for this NNI interface:

1 Open a Frame Relay profile.

2 Assign the profile a name and activate it.

```
Ethernet
   Frame Relay
      Name=ATT-NNI
      Active=Yes
```

3 Set the FR Type to NNI.

```
      FR Type=NNI
```

4 Set up the nailed connection to the remote switch and specify the data service for the link. For example:

```
      Call Type=Nailed
      Nailed Grp=1
      Data Svc=64k
```

5 Specify the link management protocol and its configuration parameters as directed by your Frame Relay provider. For example:

```
      Link Mgmt=T1.617D
      N391=6
      T391=10
```

```
T392=15
MRU=1532
```

6 Close the Frame Relay profile.

Configuring a UNI-DCE interface

In the following example, the Pipeline has a nailed connection to customer premises equipment (CPE) and will be configured with an UNI-DCE interface to that equipment. The sample network connection looks like this:

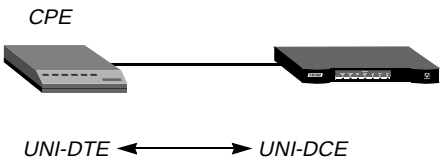


Figure 3-6. Example UNI-DCE interface to an end-point (DTE)

To configure the Frame Relay profile for this UNI-DCE interface:

1 Open a Frame Relay profile.

2 Assign the profile a name and activate it.

```
Ethernet
   Frame Relay
      Name=ATT-DCE
      Active=Yes
```

3 Set the FR Type to DCE.

```
FR Type=DCE
```

4 Set up the nailed connection to the remote switch and specify the data service for the link. For example:

```
Call Type=Nailed
Nailed Grp=1
Data Svc=64k
```

5 Specify the link management protocol and its configuration parameters as directed by your Frame Relay provider. For example:

```
Link Mgmt=T1.617D
DCE N392=3
DCE N393=4
T392=15
```

6 Close the Frame Relay profile.

Configuring a UNI-DTE interface

In this example, the Pipeline has a nailed connection to a Frame Relay switch configured as a DCE and will be configured with a UNI-DTE interface to that switch. The sample network connection looks like this:

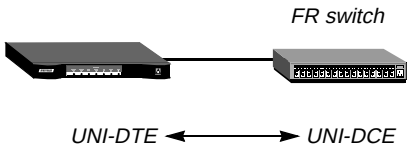


Figure 3-7. UNI-DTE interface to a Frame Relay switch

To configure the Frame Relay profile for this UNI-DTE link:

1 Open a Frame Relay profile.

2 Assign the profile a name and activate it.

```
Ethernet
   Frame Relay
      Name=ATT-DTE
      Active=Yes
```

3 Set the FR Type to DTE.

```
FR Type=DTE
```

4 Set up the nailed connection to the remote switch and specify the data service for the link. For example:

```
Call Type=Nailed
Nailed Grp=1
Data Svc=64k
```

5 Specify the link management protocol and its configuration parameters. For example:

```
Link Mgmt=Q.933A
N391=6
DTE N392=3
DTE N393=4
T391=10
```

6 Close the Frame Relay profile.

Configuring Connection profiles for Frame Relay

You must configure a Connection profile that is used in combination with the Frame Relay profile. You must specify the Frame Relay profile name as the datalink between the Pipeline and the Frame Relay network.

Understanding the Frame Relay connection parameters

This section provides some background information about configuring a Connection profile that is used in combination with the Frame Relay connection. (For more detailed descriptions of the parameters, see *Pipeline 220 VT100 Interface Reference Guide*.)

Gateway connections

Gateway connections require FR encapsulation, a Frame Relay profile name, and a DLCI. Your Frame Relay provider gives you the DLCI to assign to each connection.

The far end specified in a Frame Relay-encapsulated Connection profile lies at the end of a PVC, whose first hop is known by the DLCI named in the Connection profile. The Pipeline does not allow you to enter duplicate DLCIs, except when they are carried by separate physical links specified in different Frame Relay profiles.

Frame Relay circuits

A circuit is a PVC segment configured in two Connection profiles. Data coming in on the DLCI configured in one Connection profile is switched to the DLCI configured in the other. Data is dropped if the circuit has only one DLCI. If more than two Connection profiles specify the same circuit name, only two of them are used.

In a circuit, both Connection profiles must specify FR_CIR encapsulation and the same circuit name. Each profile must specify a unique DLCI. The Pipeline does not allow you to enter duplicate DLCIs, except when they are carried by separate physical links specified in different Frame Relay profiles.

Example connection configurations

This section shows example Connection profile configurations for Frame Relay gateway and circuit configurations.

Configuring a Frame Relay gateway connection

The following example shows you how to configure a Frame Relay gateway connection. It presumes that dial-in users who need to reach the distant IP network have valid Connection profile. This example shows the Connection profile that assigns a DLCI and passes the data stream out to a Frame Relay switch. The example network is shown in Figure 3-8:

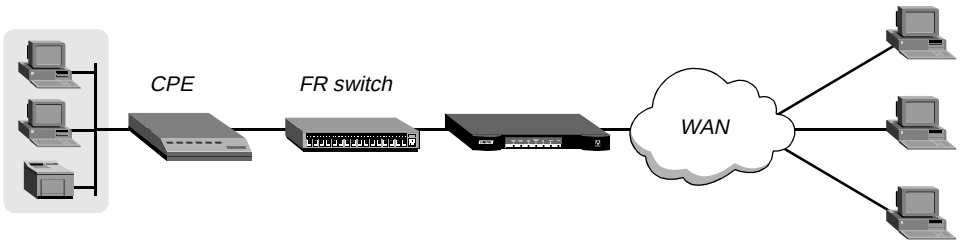


Figure 3-8. Gateway connections

In the following example, the Pipeline communicates with a remote Frame Relay switch using a Frame Relay profile named “ATT-NNI.” To configure its corresponding Connection profile:

1 Open a Connection profile.

2 Specify the station name, activate the profile, and specify FR encapsulation.

```
Ethernet
   Connections
      Station=gateway-1
      Active=Yes
      Encaps=FR
```

3 Enable IP routing and specify the address of the remote IP router.

```
Route IP=Yes
Ip options...
   LAN Adrs=10.2.3.4/24
```

4 Open the Encaps Options subprofile and specify the name of the Frame Relay profile with a nailed connection to the frame relay switch, and a DLCI assigned by the frame relay administrator.

```
Encaps options...
   FR Prof=ATT-NNI
   DLCI=55
   Circuit=N/A
```

5 Close the Connection profile.

Configuring a Frame Relay circuit

This example configuration configures a circuit between a UNI-DCE and NNI datalinks. A circuit between any two interfaces within the Pipeline would be configured in much the same way. The example network looks like this:

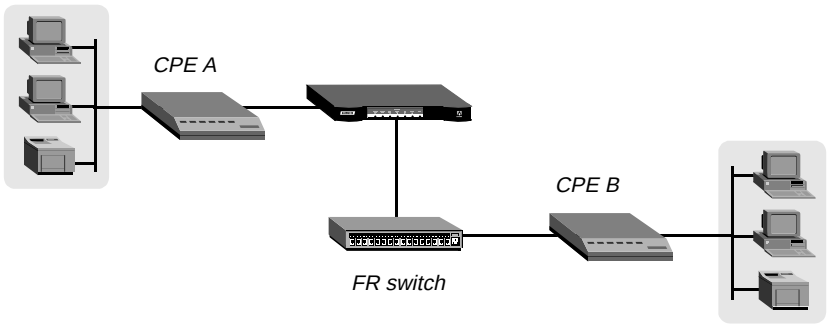


Figure 3-9. A Frame Relay circuit

The Frame Relay profile for the UNI-DCE interface in the Pipeline is named “ATT-DCE.” For the NNI interface, the Frame Relay profile is named “ATT-NNI.”

Configuring the Connection profile for the UNI-DCE interface

To configure the first Connection profile for this circuit:

1 Open the first Connection profile.

2 Specify the station name, activate the profile, and specify FR_CIR encapsulation.

```
Ethernet
   Connections
      Station=victor
      Active=Yes
      Encaps=FR_CIR
```

3 Open the Encaps Options subprofile and specify the name of the Frame Relay profile with a nailed connection to the frame relay switch, a DLCI assigned by the frame relay administrator, and a name for the frame relay circuit.

```
Encaps options...
   FR Prof=ATT-DCE
   DLCI=18
   Circuit=Circuit-1
```

4 Close the Connection profile.

Configuring the Connection profile for the NNI interface

To configure the second Connection profile for this circuit:

1 Open the second Connection profile.

2 Specify the station name, activate the profile, and specify FR_CIR encapsulation.

```
Ethernet
   Connections
      Station=marty
      Active=Yes
      Encaps=FR_CIR
```

3 Open the Encaps Options subprofile and specify the name of the Frame Relay profile with a nailed connection to the frame relay switch, a DLCI assigned by the frame relay administrator, and a name for the frame relay circuit. This name must match for both Connection profiles you create for this circuit.

```
      Encaps options...
         FR Prof=ATT-NNI
         DLCI=23
         Circuit=Circuit-1
```

4 Close the second Connection profile.

Configuring IP Routing

4

This chapter covers the following topics:

Introduction to IP routing and interfaces

The first task described in this chapter, setting up the IP network, involves setting parameters in the Pipeline unit's Ethernet profile. The parameters define the unit's Ethernet IP interface, network services (such as DNS), and routing policies.

In the next task, configuring IP routing connections, you configure Connection profiles (or similar profiles in an external authentication server) to define destinations across WAN interfaces and add routes to the routing table.

For configuring IP routes and preferences and configuring the Pipeline for dynamic route updates, you configure the IP profile and individual Connection profiles to set up the IP routing table, which determines the paths over which IP packets are forwarded and specifies the connections to be brought up.

To perform the tasks described in this chapter, you have to understand how the Pipeline uses IP addresses and subnet masks, IP routes, and IP interfaces.

For information about monitoring and managing IP routing, see the system administration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

IP addresses and subnet masks

In the Pipeline, you specify IP addresses in dotted decimal format (not hexadecimal). If you specify no subnet mask, the Pipeline assumes a default mask on the basis of address class. The

default subnet mask is the default number of network bits for the address's class. Table 4-1 shows the classes and the default number of network bits for each class.

Table 4-1. IP address classes and default subnet masks

Class	Address range	Network bits
Class A	0.0.0.0 — 127.255.255.255	8
Class B	128.0.0.0 — 191.255.255.255	16
Class C	192.0.0.0 — 223.255.255.255	24

For example, a class C address such as 198.5.248.40 has 24 network bits, so its default mask is *24*. The 24 network bits leave 8 bits for the host portion of the address. So one class C network can support up to 253 hosts.

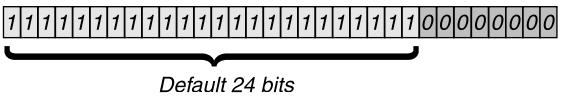


Figure 4-1. A class C IP address

For specifying a different subnet mask, the Pipeline supports a modifier that specifies the total number of network bits in the address. For example:

```
IP address = 198.5.248.40
Mask = 255.255.255.248
```

In the example address shown above, the mask specification indicates that 29 bits of the address will be used to specify the network. This is commonly referred to as a 29-bit subnet. The three remaining bits specify unique hosts.

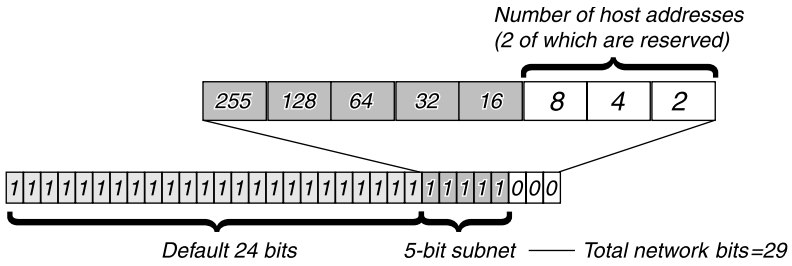


Figure 4-2. A 29-bit subnet mask and number of supported hosts

Three available bits allow eight possible bit combinations. Of the eight possible host addresses, two are reserved, as follows:

000 — Reserved for the network (base address)
001
010
100
110
101

011
111 — Reserved for the broadcast address of the subnet

Zero subnets

Early implementations of TCP/IP did not allow zero subnets. That is, subnets could have the same base address that a class A, B, or C network would have. For example, the subnet 192.168.8.0/30 was illegal because it had the same base address as the class C network 192.168.8.0/24, while 192.168.8.4/30 was legal (192.168.8.0/30 is called a zero subnet, because like a class C base address, its last octet is zero). Modern implementations of TCP/IP allow subnets to have base addresses that might be identical to the class A, B, or C base addresses. Ascend's implementations of RIP 2 and OSPF treat these so-called zero subnetworks the same as any other network. You should decide whether or not to support and configure zero subnetworks for your environment. If you configure them in some cases and treat them as unsupported in other cases, you will encounter routing problems.

Table 4-2 shows how the standard subnet address format relates to Ascend notation for a class C network number.

Table 4-2. Standard subnet masks

Subnet mask	Number of host addresses
255.255.255.0	254 hosts + 1 broadcast, 1 network (base)
255.255.255.128	126 hosts + 1 broadcast, 1 network (base)
255.255.255.192	62 hosts + 1 broadcast, 1 network (base)
255.255.255.224	30 hosts + 1 broadcast, 1 network (base)
255.255.255.240	14 hosts + 1 broadcast, 1 network (base)
255.255.255.248	6 hosts + 1 broadcast, 1 network (base)
255.255.255.252	2 hosts + 1 broadcast, 1 network (base)
255.255.255.254	invalid netmask (no hosts)
255.255.255.255	1 host — a host route

The broadcast address of any subnet has the host portion of the IP address set to all ones. The network address (or base address) represents the network itself, with the host portion of the IP address set to all zeros. Therefore, these two addresses define the address range of the subnet. For example, if the Pipeline configuration assigns the following address to a remote router:

```
IP address = 198.5.248.120
Mask = 255.255.255.248
```

The Ethernet attached to that router has the following address range:

```
198.5.248.120 – 198.5.248.127
```

A host route is a special case IP address with a subnet mask of 32 bits. It has a subnet mask of `255.255.255.255`.

IP routes

At system startup, the Pipeline builds an IP routing table that contains configured routes. When the system is up, it can use routing protocols such as RIP or OSPF to learn additional routes dynamically.

In each routing table entry, the Destination field specifies a destination network address that may appear in IP packets, and the Gateway field specifies the address of the next-hop router to reach that destination.

How the Pipeline uses the routing table

The Pipeline relies on the routing table to forward IP packets, as follows:

- If the Pipeline finds a routing table entry whose Destination field matches the destination address in a packet, it routes the packet to the specified next-hop router, whether through its WAN interface or through its Ethernet interface.
- If the Pipeline does not find a matching entry, it looks for the Default route, which is identified in the routing table by a destination of 0.0.0.0. If that route has a specified next-hop router, it forwards the packet to that router.
- If the Pipeline does not find a matching entry or does not have a valid Default route, it drops the packet.

Static and dynamic routes

A static route is a manually configured path from one network to another, which specifies the destination network and the gateway (router) to use to get to that network.

- Each Static Rtes profile specifies one static route. If a path to a destination must be reliable, the administrator often configures more than one path (that is, specifies one or more secondary routes), in which case the Pipeline chooses the route on the basis of assigned metrics and availability.
- The Ethernet>Mod Config profile specifies a static connected route, which states “to reach system-A, send packets out this interface to system-A.” Connected routes are low cost, because no remote connection is involved.
- Each IP-routing Connection profile specifies a static route that states “to reach system-A, send packets out this interface to system-B”, where system-B is another router.

A dynamic route is a path, to another network, that is learned from another IP router rather than configured in one of the Pipeline’s local profiles. Routers that use RIP broadcast their entire routing table every 30 seconds, updating other routers about the usability of particular routes. Hosts that run ICMP can also send ICMP Redirects to offer a better path to a destination network. OSPF routers propagate link-state changes as they occur. Routing protocols such as RIP and OSPF all use some mechanism to propagate routing information and changes through the routing environment.

Route preferences and metrics

The Pipeline supports route preferences, because different protocols have different criteria for assigning route metrics. For example, RIP is a distance-vector protocol, which uses a virtual hop count to select the shortest route to a destination network. OSPF is a link-state protocol, which means that OSPF can take into account a variety of link conditions, such as the reliability or speed of the link, when determining the best path to a destination network.

When choosing a route to put into the routing table, the router first compares preference values, preferring the lowest number. If the preference values are equal, the router compares the metric fields and uses the route with the lowest metric. Following are the preference values for the various types of routes:

- Connected routes have a default preference of 0.
- OSPF routes have a default preference of 10.
- ICMP redirects have a default preference of 30.
- RIP routes have a default preference of 100.
- Static routes have a default preference of 100.
- ATMP, PPTP routes have a default preference of 100.

Pipeline IP interfaces

The Pipeline must have at least one system-based IP interface (on Ethernet) to support IP routing. It also creates several internal interfaces at system startup.

At system startup, the Pipeline creates its Ethernet and internal IP interfaces.

Ethernet interfaces

The following example displays the routing table for a Pipeline configured to enable IP routing:

```
** Ascend Pipeline Terminal Server **

ascend% iproute show

Destination        Gateway   IF       Flg   Pref   Met     Use     Age
10.10.0.0/16       -         ie0      C        0     0       3     222
10.10.10.2/32      -         local    CP       0     0       0     222
127.0.0.0/8        -         bh0      CP       0     0       0     222
127.0.0.1/32       -         local    CP       0     0       0     222
127.0.0.2/32       -         rj0      CP       0     0       0     222
224.0.0.0/4        -         mcast    CP       0     0       0     222
224.0.0.1/32       -         local    CP       0     0       0     222
224.0.0.2/32       -         local    CP       0     0       0     222
224.0.0.5/32       -         local    CP       0     0       0     222
224.0.0.6/32       -         local    CP       0     0       0     222
224.0.0.9/32       -         local    CP       0     0       0     222
255.255.255.255/32 -         ie0      CP       0     0       0     222
```

The Ethernet interface has the IP address 10.10.10.2 (with a subnet mask of 255.255.0.0). No Connection profiles or static routes are configured.

Following are descriptions of the interfaces created at startup:

- The Ethernet IP interface, labeled `ie0`, is always active, because it is always connected. Its IP address is assigned in the Ethernet > Mod Config >Ether1 Options or Ether2 Options).

  The Pipeline creates two routing table entries: one with a destination of the network (labeled `ie0`), and the other with a destination of the Pipeline (labeled `local`).

  Note: Remember that the Pipeline has two separate Ethernet ports. Each of those ports support two separate IP addresses.
- The black-hole (`bh0`) interface is always up. The black-hole address is 127.0.0.3. Packets routed to this interface are discarded silently.
- The loopback (labeled `local`) interface is always up. The loopback address is 127.0.0.1/32.
- The reject (labeled `rj0`) interface is always up. The reject address is 127.0.0.2. Packets routed to this interface are sent back to the source address with an ICMP "host unreachable" message.
- Multicast interfaces have a destination address with a value of 224 for the first octet. For information about multicast addresses, see Chapter 7, "Setting Up IP Multicast Forwarding."
- Not shown in the example is an inactive interface. It is created when you configure a Connection profile. The inactive interface is where all routes point when their WAN connections are down. The inactive interface label is `wanidle0`.

WAN IP interfaces

WAN interfaces are created as they are brought up. WAN interfaces are labeled wan*N*, where *N* is a number assigned in the order in which the interfaces become active. The WAN IP address can be a local address assigned dynamically when the caller logs in, an address on a subnet of the local network, or a unique IP network address for a remote device.

Numbered interfaces

The Pipeline can operate as a both a system-based and interface-based router. Interface-based routing uses numbered interfaces. Some routers or applications require numbered interfaces, and some sites use them for trouble-shooting leased point-to-point connections and forcing routing decisions between two links going to the same final destination. More generally, interface-based routing allows the Pipeline to operate in much the same way as a multihomed Internet host.

Figure 4-3 shows a sample interface-based routing connection.

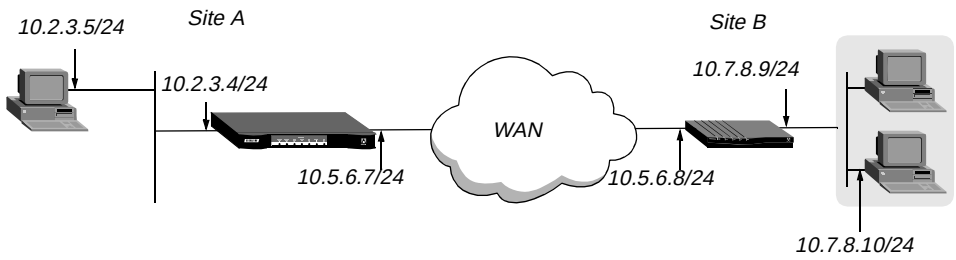


Figure 4-3. Interface-based routing example

The IP addresses 10.5.6.7 and 10.5.6.8 are assigned to the WAN interfaces. The site A Pipeline routes packets to the remote network 10.7.8.0 by means of the addresses assigned the WAN interfaces.

With system-based routing, these addresses are not assigned. The site A Pipeline routes packets to the remote network on the basis of the WAN interface it created when the connection was brought up, rather than a configured IP address.

Interface-based routing means that in addition to the system-wide IP configuration, the Pipeline and the far end of the link have link-specific IP addresses, for which you specify the following parameters:

- Connections>IP Options>IF Adrs (the link-specific address for the MAX)
- Connections>IP Options>WAN Alias (the far end link-specific address)

Or, you may omit the remote side's system-based IP address from the Connection profile and use interface-based routing exclusively. This is an appropriate mechanism, for example, if the remote system is on a backbone net that might be periodically reconfigured by its administrators, and you want to refer to the remote system only by its mutually agreed-upon interface address. In this case, the link-specific IP addresses are specified in the following parameters:

- Connections>IP Options>IF Adrs (the near end numbered interface)
- Connections>IP Options>LAN Adrs (the far end numbered interface)

Note that LAN Adrs must always be filled in, so if the only known address is the interface address, it must be placed in the Lan Adrs parameter rather than the WAN Alias parameter. In this case, a host route is created to the LAN Adrs (interface) address, a net route is created to the subnet of the remote interface, and incoming calls must report their IP addresses as the LAN Adrs address.

It is also possible, although not recommended, to specify the local numbered interface (IF Adrs) and use the far end device's system-wide IP address (LAN Adrs). In this case, the remote interface must have an address on the same subnet as the local, numbered interface.

If a Pipeline is using a numbered interface, note the following differences and similarities in operation, compared to unnumbered (system-based) routing:

- IP packets generated in the Pipeline and sent to the remote address will have an IP source address corresponding to the numbered interface, not the system-wide (Ethernet) address.
- The Pipeline adds all numbered interfaces to its routing table as host routes.
- The Pipeline accepts IP packets addressed to a numbered interface, considering them to be destined for the Pipeline itself. (The packet may actually arrive over any interface, and the numbered interface corresponding to the packet's destination address need not be active.)

Configuring the local IP network setup

The Ethernet profile configures system-global parameters that affect all IP interfaces in the Pipeline.

Understanding the IP network parameters

This section provides some background information about the IP network configuration. The information is organized by functionality rather than by parameter.

Primary IP address for each Ethernet interface

The IP Adrs parameter specifies the Pipeline unit's IP address for each local Ethernet interface. When specifying IP addresses for the Pipeline's Ethernet interfaces, you must specify the subnet mask. IP address and subnet mask are required settings for the Pipeline to operate as an IP router.

Second IP address for each Ethernet interface

The Pipeline can assign two unique IP addresses to *each* physical Ethernet port and route between them. This feature, referred to as *dual IP,* can give the Pipeline a logical interface on two networks or subnets on the same backbone.

Usually, devices connected to the same physical wire all belong to the same IP network. With dual IP, a single wire can support two separate IP networks, with devices on the wire assigned to one network or the other and communicating by routing through the Pipeline.

Dual IP is also used to distribute the load of routing traffic to a large subnet, by assigning IP addresses on that subnet to two or more routers on the backbone. When the routers have a direct connection to the subnet as well as to the backbone network, they route packets to the subnet and include the route in their routing table updates.

Dual IP also allows you to make a smooth transition when changing IP addresses. That is, a second IP address can act as a placeholder while you are making the transition in other network equipment.

Figure 4-4 shows an example IP network to which a Pipeline is connected:

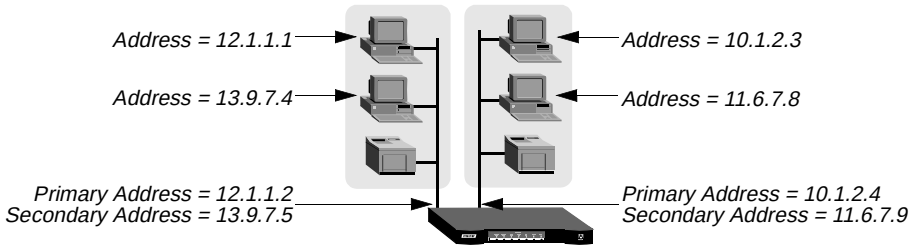


Figure 4-4. Sample dual IP network

Two IP addresses are assigned to each of the Pipeline's Ethernet interfaces. 10.1.2.4 and 11.6.7.9 are assigned to Ethernet 1. 12.1.1.2 and 13.9.7.5 are assigned to Ethernet 2. In this example, the Pipeline routes between all displayed networks. The Pipeline enables the host

assigned 12.1.1.1 to communicate with the host assigned 13.9.7.4 and the host assigned 10.1.2.3.

The host assigned 12.1.1.1 and the host assigned 13.9.7.4 share a physical cable segment, but cannot communicate unless the Pipeline routes between the 12.0.0.0 network and the 13.0.0.0 network.

Enabling RIP on the Ethernet interface

You can configure each IP interface to send RIP updates (informing other local routers of its routes), receive RIP updates (learning about networks that can be reached via other routers on the Ethernet), or both.

Note: Ascend recommends that you run RIP version 2 (RIP-v2) if possible. You should not run RIP-v2 and RIP-v1 on the same network in such a way that the routers receive each other's advertisements. RIP-v1 does not propagate subnet mask information, and the default-class network mask is assumed, while RIP-v2 handles subnet masks explicitly. Running the two versions on the same network can result in RIP-v1 class subnet mask assumptions overriding accurate subnet information obtained via RIP-v2.

Ignoring the default route

You can configure the Pipeline to ignore default routes advertised by routing protocols. This configuration is recommended, because you typically do not want the default route changed by a RIP update. The default route specifies a static route to another IP router, which is often a local router such as an Ascend GRF400 or other kind of LAN router. When the Pipeline is configured to ignore the default route, RIP updates do not modify the default route in the Pipeline routing table.

Proxy ARP and inverse ARP

The Pipeline can be configured to respond to ARP requests for remote devices that have been assigned an address dynamically. It responds to the ARP request with its own MAC address while bringing up the connection to the remote device. This feature is referred to as Proxy ARP.

The Pipeline also supports Inverse Address Resolution Protocol (Inverse ARP). Inverse ARP allows the Pipeline to resolve the protocol address of another device when the hardware address is known. The Pipeline does not issue any Inverse ARP requests, but it does respond to Inverse ARP requests that have the protocol type of IP (8000 hexadecimal), or in which the hardware address type is the two-byte Q.922 address (Frame Relay). All other types are discarded. The Inverse ARP response packet sent by the Pipeline includes the following information:

- ARP source-protocol address is the Pipeline unit's IP address on Ethernet.
- ARP source-hardware address is the Q.922 address of the local DLCI.

For the details of Inverse ARP, see RFCs 1293 and 1490.

Telnet password

The Telnet password is required from all users attempting to access the Pipeline unit via Telnet. Users are allowed three tries to enter the correct password, after which the connection attempt fails.

BOOTP relay

By default, a Pipeline does not relay BOOTP (Bootstrap Protocol) requests to other networks. If BOOTP is enabled, the Pipeline can relay BOOTP requests to another network. However, SLIP BOOTP must be disabled in Ethernet>Mod Config>TServ Options. SLIP BOOTP makes it possible for a computer connecting to the MAX over a SLIP connection to use the Bootstrap Protocol. A Pipeline can support BOOTP on only one connection. If both SLIP BOOTP and BOOTP relay are enabled, you will receive an error message.

You can specify the IP address of one or two BOOTP servers. You are not required to specify a second BOOTP server.

Note: If you specify two BOOTP servers, the MAX that relays the BOOTP request determines when each server is used. The order of the BOOTP servers in the BOOTP Relay menu does not necessarily determine which server is tried first.

Local domain name

The Pipeline uses the Domain Name parameter for DNS lookups. When the Pipeline receives a hostname to look up, it tries various combinations including appending the configured domain name. The secondary domain name (Sec Domain Name) can specify another domain name that the Pipeline can search through DNS. The Pipeline searches the secondary domain only after the domain specified in the Domain Name parameter.

DNS or WINS name servers

When the Pipeline is informed about DNS (or WINS), Telnet and Rlogin users can specify hostnames instead of IP addresses. If you configure a primary and secondary name server, the secondary server is accessed only if the primary server is inaccessible.

DNS lists

DNS can return multiple addresses for a hostname in response to a DNS query, but it does not include information about availability of those hosts. Users typically attempt to access the first address in the list. If that host is unavailable, the user must try the next host, and so forth. However, if the access attempt occurs automatically as part of immediate services, the physical connection is torn down when the initial connection fails. To avoid tearing down physical links when a host is unavailable, you can set the List Attempt parameter to allow multiple attempts before terminating the WAN session. The List Size parameter specifies the maximum number of hosts listed (up to 35).

Client DNS

Client DNS configurations define DNS-server addresses presented to WAN connections during IPCP negotiation. The configurations provide a way to protect your local DNS information from WAN users. Client DNS has two levels: a global configuration that applies to

all PPP connections (defined in the Ethernet profile), and a connection-specific configuration that applies only to the WAN connection defined in the Connection profile. The global client addresses are used only if none are specified in the connection.

SNTP service

The Pipeline can use Simple Network Time Protocol (SNTP), defined in RFC 1305, to set and maintain its system time by communicating with an SNTP server. SNTP must be enabled for the Pipeline to communicate by means of that protocol. In addition, you must specify your time zone as an offset from the Universal Time Configuration (UTC). UTC is in the same time zone as Greenwich Mean Time (GMT), and the offset is specified in hours, using a 24-hour clock. Because some time zones, such as Newfoundland, cannot use an even-hour boundary, the offset includes four digits and is stated in half-hour increments. For example, in Newfoundland the time is 1.5 hours ahead of UTC, and is represented as follows:

```
UTC +0130
```

For San Francisco, which is 8 hours ahead of UTC, you would enter:

```
UTC +0800
```

For Frankfurt, which is 1 hour behind UTC:

```
UTC -0100
```

Specifying SNTP server addresses

The Host parameter lets you specify up to three server addresses. The Pipeline will attempt to communicate with the first address. It will attempt the second only if the first is inaccessible, and the third only if the second is inaccessible.

UDP checksums

If data integrity is of the highest concern for your network, and having redundant checks is important, you can set the UDP Chksum parameter to generate a checksum whenever a UDP packet is transmitted. UDP packets are transmitted for queries and responses related to ATMP, SYSLOG, DNS, ECHOSERV, RADIUS, TACACS, RIP, SNTP, and TFTP.

Selecting UDP checksums could cause a slight decrease in performance, but in most environments the decrease is not noticeable.

Poisoning dialout routes in a redundant configuration

If you have another Ascend unit backing up the Pipeline in a redundant configuration on the same network, you can use the Adv Dialout Routes parameter to instruct the Pipeline to stop advertising IP routes if its trunks are in the alarm condition. Otherwise, it continues to advertise its dialout routes, which prevents the redundant unit from taking over the routing responsibility.

Examples of IP network configurations

This section shows an example of a simple system-IP configuration for the Ethernet interface of the Pipeline, and a more complete example with system, route, and connection configurations that work together.

Configuring the Pipeline IP interface on a subnet

On a large corporate backbone, many sites configure subnets to increase the network address space, segment a complex network, and control routing in the local environment. For example, suppose the main backbone IP network is 10.0.0.0, and it supports an Ascend GRF router at 10.0.0.17, as shown in Figure 4-5.

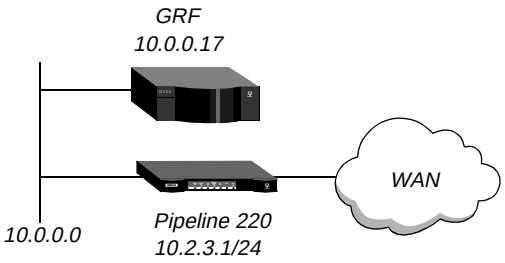


Figure 4-5. Creating a subnet for the Pipeline

Joining a subnet

You can place the Pipeline on a subnet of that network by entering a subnet mask in its IP address specification. For example:

You can place the Pipeline on a subnet of that network by entering a subnet mask in its IP address specification, for example:

1. Open Ethernet>Mod Config>Ether Options.
2. Specify the IP subnet address for the Pipeline on Ethernet. For example:

```
Ethernet
   Mod Config
      Ether options…
         IP Adrs=10.2.3.1/24
```

3. Configure the Pipeline to receive RIP updates from the local router (optional).

```
      RIP=Recv=v2
```

4. Close the Ethernet profile.

Making the backbone router the default route

With the subnet address shown in Figure 4-5, the Pipeline requires a static route to the backbone router on the main network. Otherwise, it can only communicate with devices on the subnets to which it is directly connected. To create the static route and make the backbone router the default route:

1. Open the Default IP Route profile.
2. Specify the IP address of a backbone router in the Gateway parameter. For example:

```
Ethernet
   Static Rtes
      Name=Default
      Active=Yes
      Dest=0.0.0.0/0
      Gateway=10.0.0.17
      Metric=1
      Preference=100
      Private=Yes
```

3 Close the Default IP Route profile.

See "Configuring IP routes and preferences" on page 4-21 for more information about IP Route profiles. To verify that the Pipeline is up on the local network, invoke the terminal server interface and enter the Ping command to a local IP address or hostname. For example:

```
ascend% ping 10.1.2.3
```

You can terminate the Ping exchange at any time by typing Ctrl-C.

DNS

The DNS configuration enables the Pipeline to use local DNS or WINS servers for lookups. In this example of DNS configuration, client DNS is not in use. Note that you can protect your DNS servers from callers by defining connection-specific (*client*) DNS servers and specifying that Connection profiles use those client servers. To configure the local DNS service:

1 Open Ethernet>Mod Config>DNS.

2 Specify the local domain name.

3 If appropriate, specify a secondary domain name.

4 Specify the IP addresses of a primary and secondary DNS server, and turn on the DNS list attempt feature.

```
Ethernet
   Mod Config
      DNS...
         Domain Name=abc.com
         Sec Domain Name=
         Pri DNS=10.65.212.10
         Sec DNS=12.20 7.23.51
         Allow As Client DNS=Yes
         Pri WINS=0.0.0.0
         Sec WINS=0.0.0.0
         List Attempt=Yes
         List Size=35
         Client Pri DNS=0.0.0.0
          Client Sec DNS=0.0.0.0
```

5 Close the Ethernet profile.

Configuring IP routing connections

When you enable IP routing and specify addresses in a Connection profile, you define an IP WAN interface. You must configure parameters in both the Answer profile and Connection profiles. Parameters located in the Answer profile enable you to configure parameters for all connections. Parameters in Connection profiles enable you to configure specific values for specific users.

In addition to configuring the Pipeline, you should make sure that remote hosts are properly configured.

Understanding the IP routing connection parameters

This section provides some background information about enabling IP routing in the Answer profile and Connection profiles. (For more detailed descriptions of the parameters, see the *Pipeline 220 VT100 Interface Reference Guide*.)

Enabling IP routing for WAN connections

Route IP in Answer>PPP Options must be set to Yes to enable the Pipeline to negotiate a routing connection.

Enabling IP routing for a WAN interface

To enable IP packets to be routed for this connection, set the Route IP parameter to Yes in the Connection profile. When IP routing is enabled, IP packets are always routed, they are never bridged.

Configuring the remote IP address

The LAN parameter specifies the IP address of the remote device. Before accepting a call from the far end, the Pipeline matches this address to the source IP address presented by the calling device. It may be one of the following values:

Value	How to specify
IP address of a router	If the remote device is an IP router, specify the address of its Ethernet interface, including its subnet mask modifier. (For background information, see “IP addresses and subnet masks” on page 4-1.) If you omit the subnet mask, the Pipeline inserts a default subnet mask that might make the entire far-end network accessible.
IP address of a remote host	If the remote device is a dial-in host running PPP software, specify its address, including a subnet mask modifier of 32.
The null address (0.0.0.0)	If the remote device is a dial-in host that will accept dynamic address assignment, leave the remote-address parameter blank.

Note: The most common cause of trouble in initially establishing an IP connection is incorrect configuration of the IP address or subnet mask specification for the remote host or calling device.

WAN alias

This is another IP address for the remote device, used for numbered interface routing. The WAN Alias will be listed in the routing table as a gateway (next hop) to the Lan Adrs. The caller must be using a numbered interface, and its interface address must agree with the WAN Alias setting.

Specifying a local IP interface address

IF Adrs is the Pipeline's IP address on its WAN interface. In some environments, routing might not work correctly if the Pipeline uses the default, which is its Ethernet interface IP Address.

Assigning metrics and preferences

To favor specific links, you can assign higher metrics to less desirable connections to the same location.

Each connection represents a static route, which has a default preference of 100. (For other preferences, see "Route preferences and metrics" on page 4-5.) For each connection, you can fine-tune the route preference or assign a completely different preference.

Private routes

The Private parameter specifies whether the Pipeline will disclose the existence of the route when queried by RIP or another routing protocol. Private routes are used internally. They are not advertised.

Configuring RIP policy on the WAN interface

You can configure each WAN connection to send RIP updates (informing other routers on that interface of its routes), receive RIP updates (learning about distant networks from the remote routers), or both.

Ascend recommends that you run RIP version 2 (RIP-v2) if possible. You should not run RIP-v2 and RIP-v1 on the same network in such a way that the routers receive each other's advertisements. RIP-v1 does not propagate subnet mask information, and the default class network mask is assumed, while RIP-v2 handles subnet masks explicitly. Running the two versions on the same network can result in RIP-v1 subnet mask assumptions overriding accurate subnet information obtained via RIP-v2.

Checking remote host requirements

IP hosts, such as UNIX systems, Windows or OS/2 PCs, or Macintosh systems, must have appropriately configured TCP/IP software. A remote host calling into the local IP network must also have PPP software.

UNIX software

UNIX systems typically include a TCP/IP stack, DNS software, and other software, files, and utilities used for Internet communication. UNIX network administration documentation describes how to configure these programs and files.

Windows or OS\2 software

PCs running Windows or OS/2 need TCP/IP networking software. The software is included with Windows 95, but you might need to purchase and install it separately if your computer has an older version of Windows or OS/2.

Macintosh software

Macintosh computers need MacTCP or Open Transport software for TCP/IP connectivity. MacTCP is included with all Apple system software including and later than Version 7.1. To see if a Macintosh has the software, the user should open the Control Panels folder and look for MacTCP or MacTCP Admin.

Software configuration

For any platform, the TCP/IP software must be configured with the host's IP address and subnet mask. If the host will obtain its IP address dynamically from the Pipeline, the TCP/IP software must be configured to allow dynamic allocation. If a DNS server is supported on your local network, you should also configure the host software with the DNS server's address.

Typically, the host software is configured with the Pipeline as its default router.

Examples of IP routing connections

This section provides sample Connection profile configurations for IP routing. These examples all assume that the Ethernet profile has been configured correctly, as described in "Configuring the local IP network setup" on page 4-8.

Configuring a router-to-router connection

In the following example, the Pipeline is connected to a corporate IP network and is to be connected to another company that has its own IP configuration. Figure 4-6 shows the network diagram.

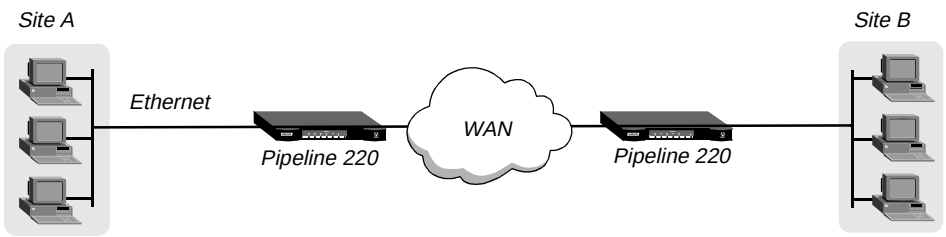


Figure 4-6. A router-to-router IP connection

This example assumes that the Answer profile in each of the two devices enables IP routing.

Configuring a Connection profile to the remote device

To configure the site A Pipeline for a connection to remote device B:

1 Open a Connection profile for the site B device.

2 Specify the remote device's name, activate the profile, and set encapsulation options.

```
Ethernet
   Connections
      Station=PipelineB
      Active=Yes
      Encaps=MPP
      Encaps options...
         Send Auth=CHAP
         Recv PW=localpw
          Send PW=remotepw
```

– Set Send PW to the value sent to the remote device to authenticate the Pipeline.

– Set Recv PW to the value the remote device sends to the Pipeline to authenticate itself.

3 Configure IP routing.

```
      Route IP=Yes
      IP options…
         LAN Adrs=10.9.8.10/22
         RIP=Off
```

4 Exit and save the Connection profile.

Configuring the remote device

To configure the site B Pipeline:

1 Open the Connection profile for the site A Pipeline.

2 Specify the site A Pipeline unit's name, activate the profile, and set encapsulation options.

```
Ethernet
   Connections
      Station=MAXA
      Active=Yes
      Encaps=MPP
      Encaps options...
         Send Auth=CHAP
         Recv PW=localpw
         Send PW=remotepw
```

3 Configure IP routing.

```
      Route IP=Yes
      IP options…
        LAN Adrs=10.2.3.1/22
         RIP=Off
```

4 Exit and save the Connection profile.

Configuring a router-to-router connection on a subnet

In this sample network, the Pipeline connects telecommuters with their own Ethernet networks to the corporate backbone. The Pipeline is on a subnet, and assigns subnet addresses to the telecommuters' networks.

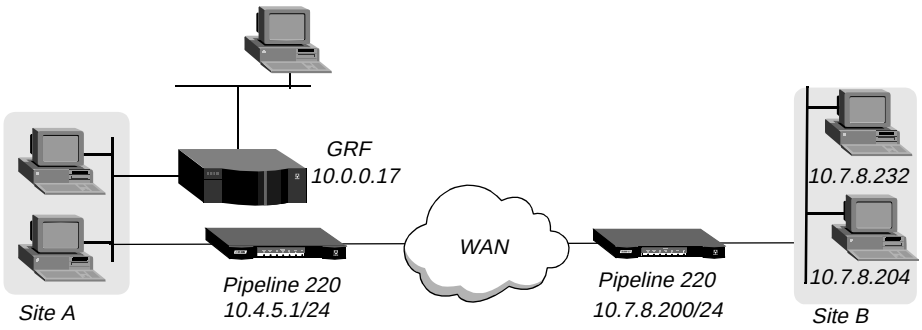


Figure 4-7. A connection between local and remote subnets

This example assumes that both of the Pipeline unit's Answer profiles enable IP routing. Because the Pipeline specifies a subnet mask as part of its own IP address, it must use other routers to reach IP addresses outside that subnet. To forward packets to other parts of the corporate network, the Pipeline either must have a default route configuration to a router in its own subnet or must enable RIP on Ethernet.

Configuring a Connection profile to the remote device

To configure the Pipeline at site A with an IP routing connection to site B:

1 Open a Connection profile for the site B device.

2 Specify the remote device's name, activate the profile, and set encapsulation options.

```
Ethernet
   Connections
      Station=PipelineB
      Active=Yes
      Encaps=MPP
      Encaps options...
         Send Auth=CHAP
         Recv PW=localpw
          Send PW=remotepw
```

– Set Send PW to the value sent to the remote device to authenticate the Pipeline.

– Set Recv PW to the value the remote device sends to the Pipeline to authenticate itself.

3 Configure IP routing.

```
Route IP=Yes
IP options…
   LAN Adrs=10.7.8.200/24
   RIP=Off
```

4 Exit and save the Connection profile.

Specifying the default route for the Pipeline

To specify the local backbone router as the Pipeline's default route:

1 Open the Default IP Route profile.

2 Specify the GRF router's address as the gateway address.

```
Ethernet
   Static Rtes
      Name=Default
      Active=Yes
      Dest=0.0.0/0
      Gateway=10.0.0.17
      Metric=1
      Preference=10
      Private=Yes
```

3 Close the IP Route profile.

Next, configure the remote router for connection to the Pipeline. Some important steps you must remember are:

- Make sure the names and passwords are correct for bi-directional authentication. The WAN link will not successfully come up if authentication fails for either the Pipeline or the remote device.
- Double-check the IP addresses to make sure they are correct.
- Be sure to configure the default route of the remote device to the IP address of the Pipeline's Ethernet interface.

Configuring a numbered interface

If you are not familiar with numbered interfaces, see "Numbered interfaces" on page 4-6. In the following example, the Pipeline is a system-based router but supports a numbered interface for one of its connections. Figure 4-8 shows an environment, that includes numbered interfaces:

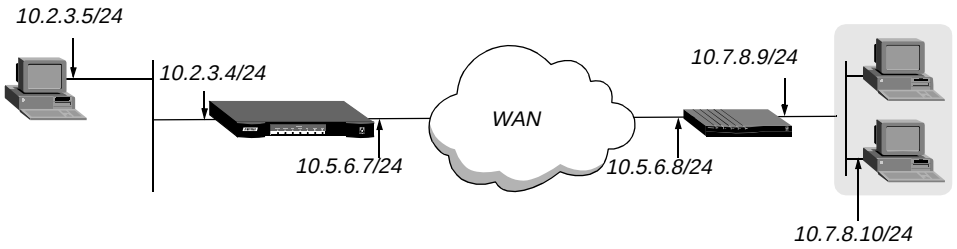


Figure 4-8. Example numbered interface

The numbered interface addresses for the Pipeline are:

- IF Adrs=10.5.6.7/24
- WAN Alias=10.5.6.8/24

To configure the numbered interface, first verify the IP address of the Pipeline, then configure the WAN connection.

Verifying the IP address of the Pipeline

To verify the IP address of the Pipeline:

1 Open Ethernet>Mod Config>Ether Options.

2 Verify that the IP Adrs parameter is set correctly for the Ethernet interface of the Pipeline:

```
IP Adrs=IP Adrs=10.2.3.4/24
Configuring the WAN connection
```

3 Close the Ethernet profile.

To configure the WAN connection and its addresses:

1 Open the Connection profile and enable IP routing.

2 Open the IP Options subprofile and specify the IP address of the remote device in the LAN Adrs parameter.

```
Ethernet
      Connections
      IP options...
         LAN Adrs=10.3.4.5/24
```

3 Specify the numbered interface address for the remote device in the WAN Alias parameter.

```
      IP options...
         WAN Alias=10.7.8.9/24
```

4 Specify the numbered interface address for the Pipeline in the IF Adrs parameter.

```
      IP options...
         IF Adrs=10.5.6.7/24
```

5 Close the Connection profile.

Configuring authentication

To configure authentication for this connection, open its Connection profile > Encaps options submenu and specify authentication method and passwords:

```
Encaps options...
   Send Auth=CHAP
   Recv PW=localpw
    Send PW=remotepw
```

– Set Send PW to the value sent to the remote device to authenticate the Pipeline.

– Set Recv PW to the value the remote device sends to the Pipeline to authenticate itself.

Configuring IP routes and preferences

The IP routing table contains routes that are configured (static routes) and routes that are learned dynamically from routing protocols such as RIP or OSPF. The section discusses static routes.

Understanding the static route parameters

This section provides some background information on static routes. For complete information about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.

Route names

IP routes are indexed by name. You can assign any name of less than 31 characters.

Activating a route

A route must be active to affect packet routing. An inactive route is ignored. You activate a route by setting Active to Yes.

Route's destination address

The Dest parameter is the destination address of a route is the target network (the destination address in a packet). Packets destined for that host use this static route to bring up the right connection. The zero address (0.0.0.0) represents the default route (the destination to which packets are forwarded when there is no route to the packet's destination).

Route's gateway address

The Gateway parameter specifies the IP address of the router or interface by which to reach the target network.

Virtual hops, costs, and preferences

The Metric parameter is a metric or hop count for the current route (a number between 1 to 15). When RIP was originally developed, the hop count was a number that showed how many routers had to be crossed to reach the destination. For example, a destination with a hop count of 10 meant that getting a packet there required going through 10 routers. A route with a shorter hop count to a destination is more desirable than one with a larger hop count, since it most likely is a shorter, faster route.

The hop count can also be manually configured to give a route a *virtual* hop count. In this way, you manually configure which routes are more desirable than others in your environment. With a choice of two identical routes but different hop count, the Pipeline uses the route with the lower hop count.

The Cost parameter specifies the cost of an OSPF link. The cost is a configurable metric that takes into account the speed of the link and other issues. The lower the cost, the more likely the interface will be used to forward data traffic. (For details, see Chapter 6, "Configuring OSPF Routing.")

The Preference parameter specifies a route preference. Zero is the default for connected routes (such as for the Ethernet interface). When choosing which route to use, the router first compares the preference values, preferring the lowest number. If the preference values are equal, the router compares the Metric values, using the route with the lowest virtual hop count. A Preference of 255 means "Don't use this route." (For more information, see "Route preferences and metrics" on page 4-5.)

Tagging routes learned from RIP

The ASE Tag value is attached to all routes learned from RIP in OSPF updates. The tag is a hexadecimal number that can be used by border routers to filter the record.

Type-1 or type-2 metrics for routes learned from RIP

The ASE Type parameter can be set to Type-1 or Type-2. Type-1 is a metric expressed in the same units as the link-state metric (that is, the same units as interface cost). Type-2 is considered larger than any link-state path. It assumes that routing between autonomous systems is the major cost of routing a packet, and eliminates the need for conversion of external costs to internal link-state metrics.

Making a route private

Private routes are used internally but are not advertised. To make a route private, set the Private Route to Yes.

Note: Typically, default routes should not be advertised to other routers. They are designed for the internal use of the specific router on which they are configured.

Routes for Connection profile interfaces

When an IP routing connection is brought up, the Pipeline activates the route for that WAN interface. The destination for the route is the remote device's IP address, and the metric and preference values are specified in the Connection profile. If the profile uses a numbered interface, an additional route is created for that interface.

A connected route for the Ethernet IP interface

The IP Adrs parameter specifies the Pipeline's IP address on the local Ethernet interface. Remember that the Pipeline has two Ethernet ports, so it has two local Ethernet interfaces. The Pipeline creates a route for these addresses at system startup.

Static route preferences

By default, static routes and RIP routes have the same preference, so they compete equally. ICMP redirects take precedence over both, and OSPF routes take precedence over everything. If a dynamic route's preference is lower than that of the static route, the dynamic route can replace, or *hide*, a static route to the same network. This can be seen in the IP routing table, which will have two routes to the same destination. The static route has an *h* flag, indicating that it is hidden and inactive. The active, dynamically learned route is also in the routing table. However, dynamic routes age, and if no updates are received, they eventually expire. In that case, the hidden static route reappears in the routing table.

RIP and OSPF preferences

Because OSPF typically involves a complex environment, its router configuration is described in a separate chapter. See Chapter 6, “Configuring OSPF Routing.”

Examples of static route configurations

This section shows two samples of static route configurations. The Pipeline forwards to the *default* route any packet that is destined for an unknown address. Following a discussion of the default route is a discussion of normal static route configuration.

Configuring the default route

If no routes exist for the destination address of a packet, the Pipeline forwards the packet to the default route. Most sites use the default route to specify a local IP router (such as a Cisco router or a UNIX host running the route daemon) to offload routing tasks to other devices.

Note: If the Pipeline does not have a default route, it drops packets for which it has no route.

The name of the default IP Route profile is always Default, and its destination is always 0.0.0.0. You cannot change these values. To configure the default route:

1 Open the first IP Route profile (the route named Default) and activate it.

```
Ethernet
   Static Rtes
      Name=Default
      Active=Yes
      Dest=0.0.0.0/0
```

2 Specify the router to use for packets with unknown destinations; for example:

```
      Gateway=10.9.8.10
```

3 Specify a metric for this route, the route’s preference, and whether the route is private. For example:

```
      Metric=1
      Preference=100
      Private=Yes
```

4 Close the IP Route profile.

Defining a static route to a remote subnet

If the connection does not enable RIP, the Pipeline does not learn about other networks or subnets that are reachable through the remote device, such as the remote network shown in Figure 4-9.

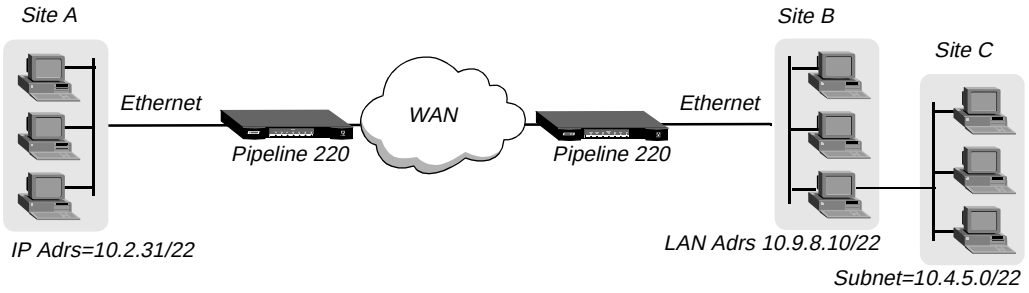


Figure 4-9. Two-hop connection that requires a static route when RIP is off

To enable the Pipeline to route to site C without using RIP, you must configure a static IP route profile:

1 Open a Static Rtes profile.

2 Enter a name for the route.

The name you choose does not affect routing. It allows you to use descriptive names for routes, in addition to their numerical representations.

3 To activate the route, set Active to Yes.

4 Specify the destination network for this route. For example:

```
Dest=10.4.5.0/22
```

5 Specify the router to use for packets destined for the network specified above. For example:

```
Gateway=10.9.8.10
```

6 Specify a metric or hop count for this route and the route's preference. For example:

```
Metric=2
Preference=100
```

7 If you use OSPF, specify values for OSPF-related parameters. For example:

```
Ospf-Cost=1
ASE-type=Type1
ASE-tag=c0000000
```

8 Exit and save the Static Rtes profile.

Example route preferences configuration

The following example increases the preference value of RIP routes, instructing the router to use a static route first if one exists.

1 Open Ethernet>Mod Config>Route Pref.

2 Set Rip Preference to 150.

```
Ethernet
   Mod Config
      Route Pref…
         Rip Preference=150
```

3 Close the Ethernet profile.

Configuring the Pipeline for dynamic route updates

Each active interface can be configured to send or receive RIP or OSPF updates. The Ethernet interface can also be configured to accept or ignore ICMP redirects. All of these routing mechanisms modify the IP routing table dynamically.

Understanding the dynamic routing parameters

This section provides some background information about the dynamic routing options. For complete information about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*. (For information about OSPF updates, see Chapter 6, "Configuring OSPF Routing.")

RIP (Routing Information Protocol)

You can configure the router to send or receive RIP updates (or both) on the Ethernet interface and on each WAN interface. The Answer profile setting applies to sessions that do not have a Connection profile. You can also choose between RIP-v1 and RIP-v2 on any interface. Many sites turn off RIP on WAN connections to keep their routing tables from becoming very large.

Note: The IETF has voted to move RIP-v1 into the *historic* category and its use is no longer recommended. Ascend recommends that you upgrade all routers and hosts to RIP-v2. If you must maintain RIP-v1, you should create a separate subnet and place all RIP-v1 routers and hosts on that subnet.

Ignoring the default route

You can configure the Pipeline to ignore default routes advertised by routing protocols. This configuration is recommended, because you typically do not want the default route changed by a RIP update. The default route specifies a static route to another IP router, which is often a local router such as an Ascend GRF 400 or other kind of LAN router. When the Pipeline is configured to ignore the default route, RIP updates do not modify the default route in the Pipeline routing table.

RIP policy and RIP summary

The RIP Policy and RIP Summary parameters have no effect on RIP-v2.

If the Pipeline is running RIP-v1, the RIP Policy parameter specifies a split horizon or poison reverse policy to handle update packets that include routes that are received on the same interface on which the update is sent. Split-horizon means that the Pipeline does not propagate routes back to the subnet from which they were received. Poison-reverse means that it propagates routes back to the subnet from which they were received, with a metric of 16.

The RIP Summary parameter specifies whether to summarize subnet information when advertising routes. If the Pipeline summarizes RIP routes, it advertises a route to all the subnets in a network of the same class. For example, the route to 200.5.8.13/28 (a class C address subnetted to 28 bits) would be advertised as a route to 200.5.8.0. When the Pipeline does not summarize information, it advertises each route in its routing table *as-is.* For the subnet in the preceding example, the Pipeline would advertise a route only to 200.5.8.13.

Ignoring ICMP redirects

ICMP was designed to dynamically find the most efficient IP route to a destination. ICMP Redirect packets are one of the oldest route-discovery methods on the Internet. They are also one of the least secure methods, because it is possible to counterfeit ICMP Redirects and change the way a device routes packets.

Private routes

If you configure a profile with Private set to Yes, the router does not disclose its route in response to queries from routing protocols.

Examples of RIP and ICMP configuration

The following sample configuration instructs the router to ignore ICMP Redirect packets, to receive (but not send) RIP updates on Ethernet, and to send (but not receive) RIP updates on a WAN connection.

Configuring RIP on a WAN link

1. Open Ethernet>Mod Config>Ether1 Options.
2. Configure the router to receive (but not send) RIP updates on Ethernet.

   ```
   Ethernet
      Mod Config
         Ether options…
            RIP=Recv-v2
   ```

   Receiving RIP updates on Ethernet means that the router will learn about networks that are reachable via other local routers. However, it will not propagate information about all of its remote connections to the local routers.
3. Close the Ether Options subprofile, and set ICMP Redirects to Ignore.

   ```
         ICMP Redirects=Ignore
   ```
4. Close the Ethernet profile.
5. Open Connections>IP Options, and configure the router to send (but not receive) RIP updates on this link.

   ```
   Ethernet
      Connections
         IP options...
            RIP=Send-v2
   ```

   Sending RIP on a WAN connection means that the remote devices will be able to access networks that are reachable via other local routers. However, the Pipeline will not receive information about networks that are reachable through the remote router.
6. Close the Connection profile.

Syslog services

To maintain a permanent log of Pipeline system events and send Call Detail Reporting (CDR) reports to a host that can record and process them, configure the Pipeline to report events to a syslog host on the local IP network. Note that syslog reports are only sent through the Ethernet interface.

Configuring the Pipeline to send Syslog messages

To configure the Pipeline to send messages to a Syslog daemon:

1. Open the Ethernet > Mod Config > Log menu.
2. Enable Syslog"

   ```
   Syslog=Yes
   ```

3. Set Syslog Host to the IP address of the host running the Syslog daemon. For example:

   ```
   Log Host=10.1.3.7
   ```

   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 Pipeline, the Pipeline must have a route to that host, either via RIP or a static route.
4. Set Syslog Port to the port number at which you want the Syslog host expects to receive messages from this Pipeline.

   ```
   Log Port=514
   ```

5. Set the Log Facility Code.

   ```
   Log Facility=Local0
   ```

   This parameter is used to flag messages from the Pipeline. After you set a Log Facility Code, you must configure the Syslog daemon to write all messages containing that facility code to a particular log file on the Syslog host.

Note: The Log Call Information parameter does not apply to the Pipeline. Setting it to either value does not affect operation of the Pipeline.

To configure the Syslog daemon, you must modify `/etc/syslog.conf` on the log host. This file specifies which action the daemon performs when it receives messages from a particular log facility number (which represents the Pipeline). For example, if you set Log Facility Code to Local5 in the Pipeline, and you want to log its messages in `/var/log/Pipeline`, add this line to `/etc/syslog.conf`:

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

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

Syslog messages

In addition to the normal traffic logged by Syslog, information may be generated for packets seen by the Secure Access firewall, if specified by SAM. By default, SAM will cause a syslog message to be generated for all packets blocked by a firewall.

Syslog messages use the following standardized format:

```
<date> <time> <router name> ASCEND: <interface> <message>
```

- <date> indicates the date the message was logged by syslog.
- <time> indicates the time the message was logged by syslog.
- <router name> indicates the router this message was sent from.
- <interface> is the name of the interface (ie0, wan0, and so on) or ‘call’ if the packet is logged by the call filter as it brings up the link.
- The <message> format has a number of fields, one or more of which may be present:

protocol — The 4 hexadecimal digit Ether Type, or the network protocol name—“arp,” “rarp,” “ipx,” “appletalk.”
The protocol for IP protocols, is either the IP protocol number (up to 3 decimal digits) or one of the following names:

- ip-in-ip
- tcp
- icmp—In the special case of icmp, it will also include the ICMP Code and Type ([Code]/[Type]/icmp).
- udp
- esp
- ah

local — For non-IP packets, is the source Ethernet MAC address of transmitted packets and the destination Ethernet MAC address of received packets. On a non-bridged WAN connection, the two MAC addresses will be all zeros.
Local for IP protocols, is the IP source address of transmitted packets and the IP destination address of received packets. In the case of TCP or UDP, it will also include the TCP or UDP port number ([IP-address];[port]).

direction — An arrow “<-”, “->” showing the direction (receive and send respectively) in which the packet was traveling.

remote — For non-IP protocols, has the same format as “local” non-IP packets but shows the destination Ethernet MAC address of transmitted packets and the source Ethernet MAC address of received packets.
For IP protocols, 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 — Used if the packet has a non-zero IP offset or the IP More-Fragments bit is set in the IP header.

log — Used to report 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 only 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.

IP Address Management

5

This chapter includes the following topics:

BOOTP Relay

The Bootstrap Protocol (BOOTP) defines how a computer on a TCP/IP network can obtain its Internet Protocol (IP) address and other startup information from another computer. The computer that requests startup information is called the BOOTP client, and the computer that supplies the startup information is called the BOOTP server. A request for startup information is called a BOOTP request, and the BOOTP server's response is called a BOOTP reply.

When the BOOTP client and BOOTP server are not on the same local-area network, the BOOTP request must be relayed from one network to another. The Pipeline can perform this task, which is known as BOOTP relay.

A device that relays BOOTP requests to another network is known as a BOOTP relay agent. In addition to delivering BOOTP requests to servers, a BOOTP relay agent is responsible for delivering BOOTP replies to clients. In most cases, the agent is a router, such as the Pipeline, that connects the networks.

By default, a Pipeline does not relay BOOTP requests to other networks. To enable BOOTP relay:

1 Obtain the IP address of up to two BOOTP servers to be used.

2 Open the Ethernet > Mod Config:

```
20-A00 Mod Config
 BOOTP Relay...
 >BOOTP Relay Enable=No
  Server=0.0.0.0
  Server=0.0.0.0
```

3 Select BOOTP Relay Enable and set it to Yes.

4 Specify the IP address of the BOOTP server.

5 If there is another BOOTP server available, enter its IP address.

You are not required to specify a second BOOTP server.

If you specify two BOOTP servers, the Pipeline that relays the BOOTP request determines when each server is used. The order of the BOOTP servers in the BOOTP Relay menu does not necessarily determine which server is tried first.

DHCP services

A Pipeline can perform a number of Dynamic Host Configuration Protocol (DHCP) services, including:

- DHCP Server functions, responding to DHCP requests for up to 20 clients at any given time. DHCP server responses provide an IP address and subnet mask.
- Managing Plug and Play requests for TCP/IP configuration settings from computers using Microsoft Windows 95 or Windows NT.
- DHCP Spoofing responses, supplying a temporary IP address for a single host. The IP address supplied is always one greater than that of the Pipeline's Ethernet interface. The IP address is good for 60 seconds. Once a session is established, an official IP address can be retrieved from a remote DHCP or BOOTP server.

How IP addresses are assigned

When a Pipeline is configured to be a DHCP server and it receives a DHCP client request, it assigns an IP address in one of the following ways:

Method	Description
Plug-and-Play	If DHCP PNP Enabled is set to Yes, the Pipeline increments its own IP address by one, and returns that address in the BOOTP reply message along with IP addresses for the Default Gateway and Domain Name Server. Plug-and-play works with Microsoft Windows 95 (and potentially other IP stacks) to assign an IP address and other wide-area networking settings to a requesting device automatically. With Plug-and-Play you can use the Pipeline to respond to distant networks without having to configure an IP address first.
Renewal	If the host is renewing the address it currently has, the Pipeline assigns the host the same address. When a host gets a dynamically assigned IP address from one of the address pools, it periodically renews the lease on the address until it has finished using it, as defined by the DHCP protocol. If the host renews the address before its lease expires, the Pipeline always provides the same address.
Next available	If the host is making a new request and there is no IP address reserved for the host, the Pipeline assigns the next available address from its address pools. The Pipeline assigns the first available address from the first pool IP addresses. If there are no available addresses in the first pool, the Pipeline assigns the first available address from the second pool.

Configuring DHCP services

To configure DHCP services, you must first enable it. Then, you configure pools of IP addresses that the Pipeline assigns to any host requesting an address. Additionally, you can configure up to six specific host addresses that receive specific IP addresses.

Enabling DHCP services

To enable DHCP services:

1. Open the Ethernet > Mod Config > DHCP Spoofing submenu.
2. Set the DHCP Spoofing parameter to Yes to enable any DHCP service.

   This parameter must be Yes for any DHCP service to be enabled. If it is set to No, other settings in this menu are ignored.
3. Set the DHCP PNP Enabled parameter to Yes to enable Plug and Plug.

   Setting this parameter to Yes and DHCP Spoofing set to Yes is all that is required to enable Plug and Play support.
4. Set Renewal Time to the amount of seconds a DHCP-assigned IP address is valid before it needs to be renewed. It applies to DHCP spoofed addresses and DHCP server replies. If the host renews the address before it expires, the Pipeline provides the same address.

   Note: Plug and Play addresses always expire in 60 seconds.
5. Enable Become Default Router to advertise the address of your Pipeline as the default router for all DHCP request packets.
6. Set Always Spoof as follows:

   – **Yes enables the DHCP server.** A DHCP server always supplies an IP address for every request, until all IP addresses are exhausted.

   – **No enables DHCP spoofing.** DHCP spoofing only supplies an IP address for a single host on the network. It does not respond to all requests.

   If both DHCP Spoofing and Always Spoof are Yes, the DHCP server feature is enabled. If DHCP Spoofing is Yes and Always Spoof is No, DHCP spoofing is enabled.

Configuring IP address pools

To configure pools of addresses from which the Pipeline assigns IP addresses to requesting hosts:

1. In the Ethernet > Mod Config > DHCP Spoofing submenu, set the IP Group 1 parameter to the first address the Pipeline uses for its first DHCP address pool.
2. Set Group 1 Count to the maximum number of addresses that the Pipeline can assign from this pool.
3. If appropriate, set the IP Group 2 parameter to the first address the Pipeline uses for its second DHCP address pool.
4. If appropriate, set Group 2 Count to the maximum number of addresses that the Pipeline can assign from this pool.
5. Exit and save the profile.

Assigning specific addresses to particular hosts

To configure specific IP addresses for use by particular hosts:

1. In the Ethernet > Mod Config > DHCP Spoofing submenu, set the IP address that the Pipeline assigns to the requesting host using the Host 1 IP parameter.
2. Specify the Ethernet (MAC) address of the host that is assigned the configured IP address using the Host 1 Enet parameter.
3. For additional hosts, repeat step 1 and step 2, using the other Host IP and Host Enet parameters. You can specify up to six hosts.

Local DNS host address table

You can create a local DNS table that provides a list of IP addresses for a specific host name when the remote DNS server fails to resolve the host name successfully. The local DNS table provides the list of IP addresses only if the host name for the attempted connection matches a host name in the local DNS table.

You create the DNS table from the terminal server by entering the host names and their IP addresses in the table. A table can contain up to eight entries, with a maximum of 35 IP addresses for each entry. You enter only the first IP address; any other IP addresses in the list are automatically added if you have enabled automatic updating of the list.

Also, you can specify that the local DNS table is automatically updated when a connection to a host whose name matches one in the local DNS table is successfully resolved by the remote DNS. When the table is updated, the returned IP address list from the remote server replaces the stored IP addresses for that host name in the local DNS list.

Configuring the local DNS table

To enable and configure the local DNS table:

1. Open the Ethernet > Mod Config > DNS menu.
2. Select List Attempt=Yes to allow a list of the IP addresses to be displayed when using the terminal server command Dnstab Entry.
3. Select List Size and enter the number of entries you want in the list.

   The minimum value is 1. The maximum value is 35.

   The number of IP addresses displayed with the Dnstab Entry command depends upon the value you set in the List Size parameter.

   If List Attempt=Yes, and the name server returns an IP address list, the list is copied into the entry in the local DNS table that matches the host name, up to the number of entries you specify in List Size. When a list of IP addresses for an entry is automatically updated, any existing list for that entry is discarded.

   For example:

   – If you set List Size=4 and the remote DNS returns 3 entries, the entire list of IP addresses in the local DNS table is cleared and the three returned addresses are entered for the entry.

   – If the local DNS table already contains 35 IP addresses for an entry and the remote DNS server returns only 4, or if you set List Size=4, the first four IP addresses are

entered into the table for the entry and the remaining addresses in the list are set to zero.

– If you set List Size=1, the list can contain only one IP address; any others returned by the remote DNS are ignored. If you change the List Size parameter value from a number greater than one to one, only the first IP address is retained; all others are set to zero the next time the table entry for that name is updated.

4 Select Enable Local DNS Table=Yes.
The default is No.

5 Select Loc DNS Tab Auto Update=Yes to enable automatic updating.
The default is No. When automatic updating is enabled, the list of IP addresses for each entry is replaced with a list from the remote DNS when the remote DNS successfully resolves a connection to a host named on the table.

For additional information, see the description of the DNSTab command in the *Pipeline 220 VT100 Interface Reference Guide*.

User-definable TCP connection retry timeout

You can set the TCP timeout parameter to the maximum length of time the Pipeline waits to complete a connection before trying the next address supplied by a DNS server. If the Pipeline cannot connect to the first host in the list, it tries the next (if available), until it connects or times out.

You should set the TCP Timeout parameter on the basis of the characteristics of the TCP destination hosts. For example, if the destinations are on a local network under the same administrative control as the Pipeline and are lightly loaded, then a short timeout (a few seconds) might be acceptable, because a host that does not respond within that interval is probably down.

A longer timeout is appropriate if the environment includes servers with:

- Longer network latency times
- High loads on the net or router
- Remote host with characteristics that are not well known.

TCP timeout values of 30 to 60 seconds are common in UNIX TCP implementations.

The default value, zero, specifies that the Pipeline waits for a maximum of 170 seconds to connect to each address on the list, until a connection is successful or the connection is dropped.

Network Address Translation (NAT)

To connect to the Internet or any other TCP/IP network, a host must have an IP address that is unique within that network. The Internet guarantees the uniqueness of addresses by creating central authorities that assign official IP addresses. However, many local networks use private IP addresses that are unique only on the local network. To allow a host with a private address to communicate with the Internet or another network that requires an official IP address, a Pipeline can be configured to use Network Address Translation (NAT). It works as follows:

- When the local host sends packets to the remote network, the Pipeline translates the host's private address on the local network to an official address on the remote network.
- When the local host receives packets from the remote network, the Pipeline translates the official address on the remote network to the host's private address on the local network.

NAT and port routing

A Pipeline performs NAT in the following ways:

- For more than one host on the local network without borrowing IP addresses from a DHCP server on the remote network.
- When the remote network initiates the Frame Relay connection to the Pipeline.
- By routing packets it receives from the remote network for up to 10 different TCP or UDP ports to specific hosts and ports on the local network.

When using NAT, the Pipeline is the only local host that is visible to the remote network.

How address translation works

When using NAT, the Pipeline replaces, with the official address, the local IP address of any packet received from the Ethernet interface. The Pipeline makes an entry in its internal translation table, and uses the table to keep track of all active NAT sessions.

When the Pipeline receives packets from the WAN, they all have the Pipeline's address as their destination. You can configure the Pipeline to route the packets to up to 10 different TCP or UDP ports on specific local servers. You must configure the list of local servers and the UDP and TCP ports each handles. When you configure the Pipeline to route incoming packets, destined for a particular TCP or UDP port, to a specific local server, multiple hosts on the remote network can connect to the server at the same time.

For example, if you configure the Pipeline to route all incoming packets for TCP port 80 (the standard port for HTTP) to port 80 of your local World Wide Web server.

The port you route to does not have to be the same as the port specified in the incoming packets. For example, you can route all packets received from the WAN and destined for TCP port 119, the well known port for Network News Transfer Protocol, to port 1119 on your local Usenet News server.

You can also define a local default server that handles UDP and TCP ports not listed. If you don't specify any routed ports but do specify a default server, the default server receives all packets that the remote network sends to the Pipeline.

The number of simultaneous connections is limited to the size of the translation table.

Translation table size

The Pipeline maintains an internal NAT translation table limited to 500 addresses. A translation table entry represents one TCP or UDP connection.

Note: A single application can generate many TCP and UDP connections.

The translation table entries are freed based on the following time-outs:

- Non-DNS UDP translations timeout after 5 minutes.
- DNS times out in one minute.
- TCP translations time out after 24 hours.

The translation table entries are reused as long as packets are seen that match an entry. All are freed (expired) when a connection disconnects. For Nailed connections, the connection is designed not to disconnect.

Configuring NAT

To configure NAT on the Pipeline:

1 Open the menu Ethernet > NAT > NAT menu.

2 Enable NAT by setting Routing to Yes. Without this setting, no other setting is valid.

3 Set Profile to the name of a Connection profile you want to use to connect to the Network Access Server (NAS).

4 FR address refers to Frame Relay. Refer to “Well-known ports” on page 5-8 for more information.

Configuring NAT port routing

You can configure a Pipeline to perform NAT for remote hosts wishing to access services on the local network in one of these ways:

- Route incoming packets from a remote network for up to 10 different TCP or UDP ports to specific servers and ports on the local network and, optionally, route any remaining packets to a default server.
- Route all incoming packets from a remote network to a single server on the local network.

If the Pipeline receives a packet of the configured type and port, it is forwarded to the local IP address and port you specify.

You can configure up to ten static mappings, directing the Pipeline where to forward packets addressed to specific ports. Each Static Mapping submenus contain parameters for controlling the routing of packets from a remote network to a specific TCP or UDP port. To configure static mappings:

1 Open a NAT> Static Mapping submenu.

Each Static Mapping *nn* menu contains the following parameters:

```
20-A00 NAT
   Static Mapping 01
      Valid=Yes
      Dst Port#=21
      Protocol=TCP
```

```
Loc Port#=21
Loc Adrs=181.100.100.102
```

2. Set Valid to Yes.
3. Set the Dst Port. Any packet received from the WAN that has been sent to the port specified in the Destination Port parameter is forwarded to the port specified in the Local Port parameter.
4. Set Protocol to the type of packets to which the Pipeline should apply this mapping.
5. Set Loc Port to the port to which the Pipeline forwards packets on the local network.
6. Set Loc Address to the IP address of the host to which the Pipeline forwards packets on the local network.
7. Optionally, set Default Server to the IP address of a local server to which the Pipeline routes incoming packets that are *not* routed to a specific server and port.
8. Exit and save the profile.

Note: If you have additional routers on your local area network, open Ethernet > Mod Config > Ether Options, and set the value of Ignore Def Rt to Yes. This avoids the possibility that a default route from the ISP will overwrite the NAT route.

Well-known ports

TCP and UDP ports numbered 0-1023 are assigned by the Internet Assigned Numbers Authority (IANA). In most cases, the TCP and UDP port numbers for a service are the same.

You can obtain current lists of well-known ports and registered ports (ports in the range 1024-4915 that are registered with the IANA) via FTP from

`ftp://ftp.isi.edu/in-notes/iana/assignments/port-numbers`

Configuring OSPF Routing

6

This chapter covers the following topics:

Introduction to OSPF

Open Shortest Path First (OSPF) is a second generation Internet routing protocol. The *Open* in its name refers to the fact that OSPF was developed in the public domain as an open specification. *Shortest Path First* refers to an algorithm developed by Dijkstra in 1978 for building a self-rooted shortest-path tree from which routing tables can be derived. The algorithm is described in "The link-state routing algorithm" on page 6-7.

For information about monitoring and managing OSPF, see the system administration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

RIP limitations solved by OSPF

The rapid growth of the Internet has pushed the Routing Information Protocol (RIP) beyond its capabilities, especially because of the following problems:

Problem	Description and solution
Distance-vector metrics	RIP is a distance-vector protocol, which uses a hop count to select the shortest route to a destination network. RIP always uses the lowest hop count, regardless of the speed or reliability of a link.
	OSPF is a link-state protocol, which means that OSPF can take into account a variety of link conditions, such as the reliability or speed of the link, when determining the best path to a destination network.
15-hop limitation	A destination that requires more than 15 consecutive hops is considered unreachable, which inhibits the maximum size of a network.
	OSPF has no hop limitation. You can add as many routers to a network as you want.

Problem	Description and solution
Excessive routing traffic and slow convergence	RIP creates a routing table and then propagates it throughout the internet of routers, hop by hop. The time it takes for all routers to receive information about a topology change is called *convergence*. A slow convergence can result in routing loops and errors. A RIP router broadcasts its entire routing table every 30 seconds. On a 15-hop network, convergence can be as high as 7.5 minutes. In addition, a large table can require multiple broadcasts for each update, which consumes a lot of bandwidth. OSPF uses a topological database of the network and propagates only changes to the database (as described in "Exchange of routing information" on page 6-4.)

Ascend implementation of OSPF

The primary goal for OSPF in this release is to allow the Pipeline to communicate with other routers within a single Autonomous System (AS).

The Pipeline acts as an OSPF internal router with limited border-router capability. For this release, you should not configure the Pipeline as an Area Border Router (ABR), so the Ethernet interface and all of the Pipeline WAN links should be configured in the same area.

The Pipeline does not function as a full AS Border Router (ASBR) at this release. However, it performs ASBR calculations for external routes such as WAN links that do not support OSPF. The Pipeline imports external routes into its OSPF database and flags them as Autonomous System External (ASE). It redistributes those routes via OSPF ASE advertisements, and propagates its OSPF routes to remote WAN routers running RIP.

The Pipeline supports null and simple password authentication.

OSPF features

This section provides a brief overview of OSPF routing to help you configure the Pipeline properly. For full details about how OSPF works, see RFC 1583, "OSPF Version 2," 03/23/1994, J. Moy.

An Autonomous System (AS) is a group of OSPF routers exchanging information, typically under the control of one company. An AS can include a large number of networks, all of which are assigned the same AS number. All information exchanged within the AS is *interior.*

Exterior protocols are used to exchange routing information between autonomous systems. The are referred to as Exterior Gateway Protocol (EGP). The AS number can be used by border routers to filter out certain EGP routing information. OSPF can make use of EGP data generated by other border routers, and added into the OSPF system as ASEs, as well as static routes configured in the Pipeline or RADIUS.

Security

All OSPF protocol exchanges are authenticated. This means that only trusted routers can participate in the AS's routing. A variety of authentication schemes can be used. In fact, different authentication types can be configured for each area. In addition, authentication provides added security for the routers that are on the network. Routers that do not have the password will not be able to gain access to the routing information, because authentication failure prevents a router from forming adjacencies.

Support for variable length subnet masks

OSPF enables the flexible configuration of IP subnets. Each route distributed by OSPF has a destination and mask. Two different subnets of the same IP network number may have different sizes (different masks). This is commonly referred to as Variable-Length Subnet Masks (VLSM), or Classless Inter-Domain Routing (CIDR). A packet is routed to the best (longest, or most specific) match. Host routes are considered to be subnets whose masks are *all ones* (0XFFFFFFFF).

Note: Although OSPF is very useful for networks that use VLSM, you should assign subnets as contiguously as possible, to prevent excessive link-state calculations by all OSPF routers on the network.

Interior Gateway Protocol (IGP)

OSPF keeps all AS-internal routing information within that AS. All information exchanged within the AS is *interior.*

An AS Border Router (ASBR) is required to communicate with other autonomous systems. To do so, it must use an External Gateway Protocol (EGP), as shown in Figure 6-1. An EGP acts as a shuttle service between autonomous systems.

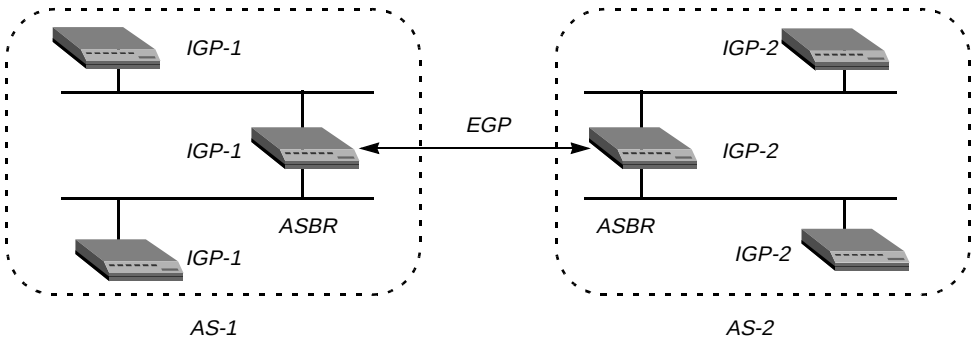


Figure 6-1. Autonomous system border routers

ASBRs perform calculations related to external routes. The Pipeline imports external routes from RIP—for example, when it establishes a WAN link with a caller that does not support OSPF—and the ASBR calculations are always performed.

Note: If you must prevent the Pipeline from performing ASBR calculations, you can disable them in Ethernet>Mod Config>OSPF Global Options.

Exchange of routing information

OSPF uses a topological database of the network and propagates only changes to the database. Part of the SPF algorithm involves acquiring neighbors, and then forming an adjacency with one neighbor, as shown in Figure 6-2.

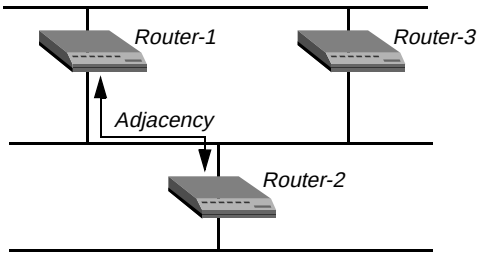


Figure 6-2. Adjacency between neighboring routers

An OSPF router dynamically detects its neighboring routers by sending its Hello packets to the multicast address All SPFRouters. It then attempts to form adjacencies with some of its newly acquired neighbors.

Adjacency is a relationship formed between selected neighboring routers for the purpose of exchanging routing information. Not every pair of neighboring routers become adjacent. Adjacencies are established during network initialization, in pairs, between two neighbors. As the adjacency is established, the neighbors exchange databases and build a consistent, synchronized database between them.

When an OSPF router detects a change on one of its interfaces, it modifies its topological database and multicasts the change to its adjacent neighbor, which in turn propagates the change to its adjacent neighbor until all routers within an area have synchronized topological databases. The result is quick convergence among routers. OSPF routes can also be summarized in Link-State Advertisements (LSAs).

Designated and backup designated routers

In OSPF terminology, a broadcast network is any network that has more than two OSPF routers attached and supports the capability to address a single physical message to all of the attached routers.

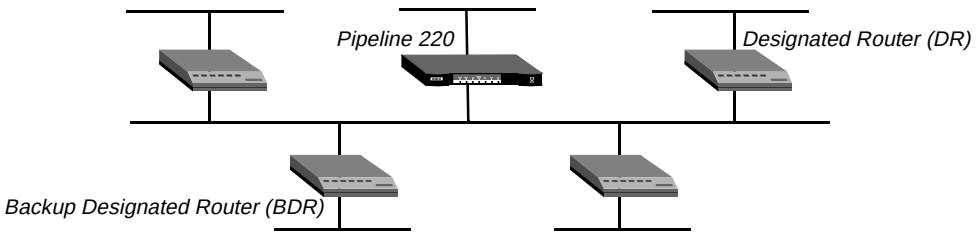


Figure 6-3. Designated and backup designated routers

The Pipeline can function as a Designated Router (DR) or Backup Designated Router (BDR). However, many sites choose to assign a LAN-based router for these roles in order to dedicate the Pipeline to WAN processing.

To reduce the number of adjacencies each router must form, OSPF calls one of the routers the designated router. A designated router is elected as routers are forming adjacencies, and then all other routers establish adjacencies only with the designated router. This simplifies the routing table update procedure and reduces the number of link-state records in the database. The designated router also plays other important roles in addition to reducing the overhead of OSPF link-state procedures. For example, other routers send link-state advertisements it to the designated router only by using the *all-designated-routers* multicast address of 224.0.0.6.

To prevent the designated router from becoming a serious liability to the network if it fails, OSPF elects a backup designated router at the same time. Other routers maintain adjacencies with both the designated router and its backup router, but the backup router leaves as many of the processing tasks as possible to the designated router. If the designated router fails, the backup immediately becomes the designated router and a new backup is elected.

The administrator chooses which router is to be the designated router on the basis of the processing power, speed, and memory of the system, and then assigns priorities to other routers on the network in case the backup designated router is also down at the same time.

Configurable metrics

The administrator assigns a cost to the output side of each router interface. The lower the cost, the more likely the interface is to be used to forward data traffic. Costs can also be associated with the externally derived routing data.

The OSPF cost can also be used for preferred path selection. If two paths to a destination have equal costs, you can assign a higher cost to one of the paths to configure it as a backup to be used only when the primary path is not available.

Figure 6-4 shows how costs are used to direct traffic over high-speed links. For example, if Router-2 in Figure 6-4 receives packets destined for Host B, it will route them through Router-1 across two T1 links (Cost=20), rather than across one 56kbps B-channel to Router-3 (Cost=240).

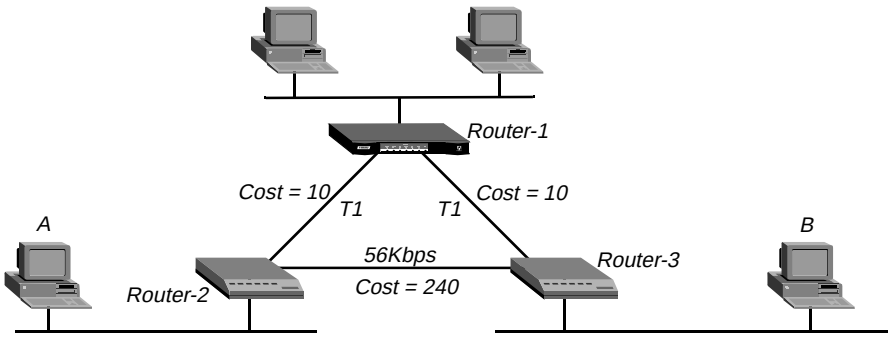


Figure 6-4. OSPF costs for different types of links

The Pipeline has a default cost of 1 for a connected route (Ethernet) and 10 for a WAN link. If you have two paths to the same destination, the one with the lower cost will be used. You

might want to reflect the bandwidth of a connection in its cost assignment. For example, for a single B-channel connection, the cost would be 24 times greater than for a T1 link.

Note: Be careful when assigning costs. Incorrect cost metrics can cause delays and congestion on the network.

Hierarchical routing (areas)

If a network is large, the size of the database, time required for route computation, and related network traffic can become excessive. An administrator can partition an AS into areas to provide hierarchical routing connected by a backbone.

The backbone area is special and always has the area number 0.0.0.0. Other areas are assigned area numbers that are unique within the autonomous system.

Each areas acts like its own network: all area-specific routing information stays within the area, and all routers within an area must have a synchronized topological database. To tie the areas together, some routers belong to an area and to the backbone area. These routers are Area Border Routers (ABRs). In Figure 6-5, all of the routers are ABRs.

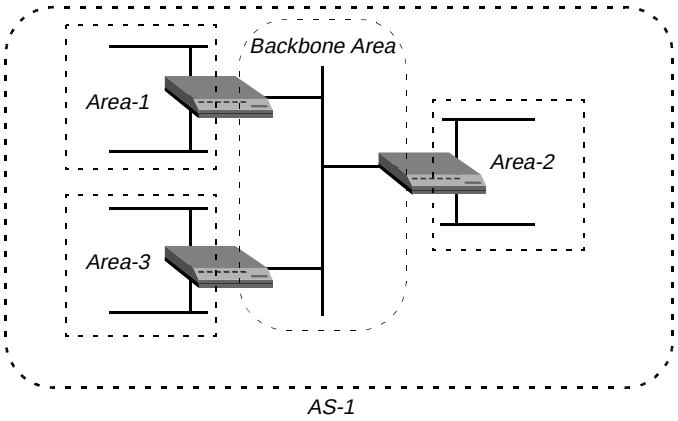


Figure 6-5. Dividing an AS into areas

If the ABRs and area boundaries are set up correctly, link-state databases are unique to an area.

Note: With this release, you should not configure the Pipeline as an ABR. You should use the same area number for the Ethernet interface of the Pipeline and each of its WAN links. That number does not have to be the backbone. The Pipeline can reside in any OSPF area.

Stub areas

To reduce the cost of routing, OSPF supports stub areas, in which all external routes are summarized by a default route. For areas that are connected to the backbone by only one ABR (that is, the area has one exit point), there is no need to maintain information about external routes. Stub areas are similar to regular areas except that the routers do not enter external routes in the area's databases.

To prevent flooding of external routes throughout the AS, you can configure an area as a stub when there is a single exit point from the area, or when the choice of exit point need not be

made on a per-external-destination basis. You might need to specify a stub area with no default cost (StubNoDefault) if the area has more than one exit point.

In a stub area, routing to AS-external destinations is based on a per-area default cost. The per-area default cost is advertised to all routers within the stub area by a border router, and is used for all external destinations.

Note: If the Pipeline supports external routes across its WAN links, you should not configure it in a stub area. Because an ABR configuration is not currently recommended for the Pipeline, the area in which it resides should not be a stub area if any of its links are AS-external.

The link-state routing algorithm

Link-state routing algorithms require that all routers within a domain maintain synchronized (identical) topological databases, and that the databases describe the complete topology of the domain. An OSPF router's domain may be an AS or an area within an AS.

OSPF routers exchange routing information and build Link-state databases. Link-state databases are synchronized between pairs of adjacent routers (as described in "Exchange of routing information" on page 6-4). In addition, each OSPF router uses its link-state database to calculate a self-rooted tree of shortest paths to all destinations, as shown in Figure 6-6.

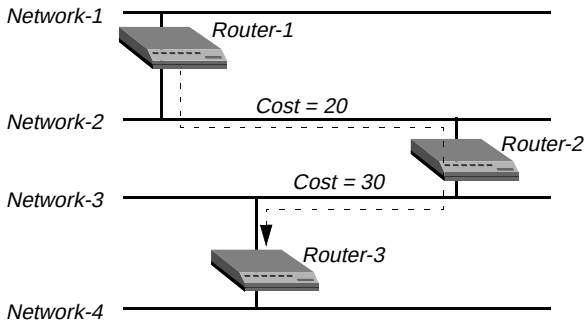


Figure 6-6. Sample network topology

The routers then use the trees to build their routing tables, as shown in Table 6-1.

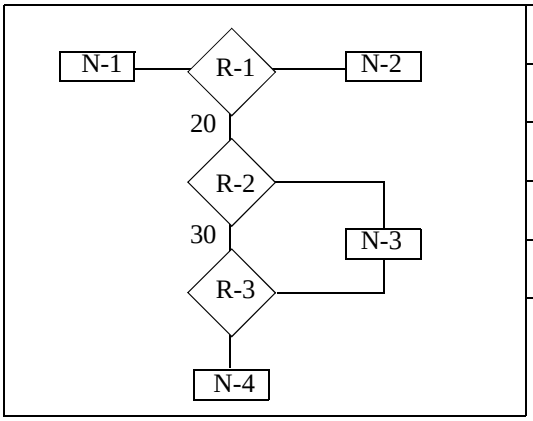
Table 6-1. Link state databases for network topology in Figure 6-6

Router-1	**Router-2**	**Router-3**
Network-1/Cost 0	Network-2/Cost0	Network-3/Cost 0
Network-2/Cost 0	Network-3/Cost0	Network-4/Cost 0
Router-2/Cost 20	Router-1/Cost 20	Router-2/Cost 30
	Router-3/Cost 30	

Table 6-2 through Table 6-4 show another example of self-rooted shortest-path trees calculated from link-state databases, and the resulting routing tables. Actual routing tables also contain

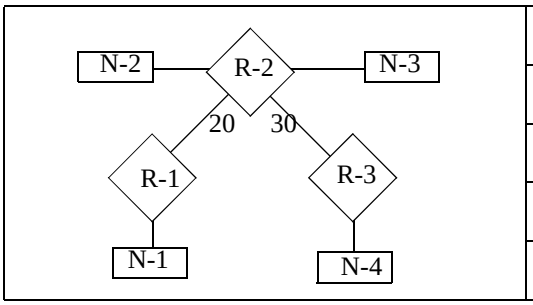
externally derived routing date, which is advertised throughout the AS but kept separate from the Link-state data. Also, each external route can be tagged by the advertising router, enabling the passing of additional information between routers on the boundary of the AS.

Table 6-2. Shortest-path tree and resulting routing table for Router-1



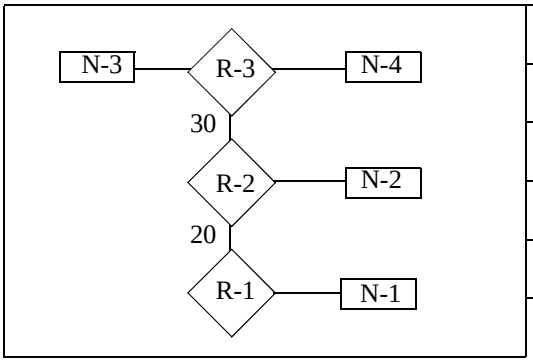
Destination	*Next Hop*	*Metric*
Network-1	Direct	0
Network-2	Direct	0
Network-3	Router-2	20
Network-4	Router-2	50

Table 6-3. Shortest-path tree and resulting routing table for Router-2



Destination	*Next Hop*	*Metric*
Network-1	Router-1	20
Network-2	Direct	0
Network-3	Direct	0
Network-4	Router-2	30

Table 6-4. Shortest-path tree and resulting routing table for Router-3



Destination	*Next Hop*	*Metric*
Network-1	Router-2	50
Network-2	Router-2	30
Network-3	Direct	0
Network-4	Direct	0

Configuring OSPF routing in the Pipeline

This section provides brief description of the OSPF routing parameters, and shows how to use the parameters in sample configurations. For additional information about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.

Understanding the OSPF routing parameters

This section provides some background information about the OSPF parameters. You can access the parameters either through the Ethernet > Mod Config > OSPF Options submenu (for global configuration of the Pipeline) or through the Ethernet > Connections > OSPF options submenu (for per-connection configuration of the Pipeline). The parameters are the same, but some of the default values are different. For OSPF routing, you configure the following settings:

Setting	Description
Enabling OSPF on an interface	OSPF is turned off by default. To enable it on an interface, set RunOSPF to Yes.
Specifying an area number and type	Area sets the area ID for the interface. The format for this ID is dotted decimal, but it is not an IP address. See “Hierarchical routing (areas)” on page 6-6.
	AreaType specifies the type of area: Normal, Stub, or StubNoDefault. See “Stub areas” on page 6-6.
Intervals for communicating with an adjacent router	HelloInterval specifies how frequently, in seconds, the Pipeline sends out Hello packets on the specified interface. OSPF routers use Hello packets to dynamically detect neighboring routers in order to form adjacencies.
	DeadInterval specifies how many seconds the Pipeline will wait before declaring its neighboring routers down after it stops receiving their Hello packets
	See “Exchange of routing information” on page 6-4.
Priority	The routers in the network use the Priority value to elect a Designated Router (DR) and Backup Designated Router (BDR). Assigning a priority of 1 would place the Pipeline near the top of the list of possible designated routers. Acting as a DR or BDR significantly increases the amount of OSPF overhead for the router. For a discussion of the functions of DRs and BDRs, see “Designated and backup designated routers” on page 6-4.
Authentication type and key	You can specify that the Pipeline supports OSPF router authentication, and the key it will look for in packets to support that authentication. See “Security” on page 6-3.
Cost of the route on this interface	This parameter specifies the link-state or output cost of a route. Assign realistic costs for each interface that supports OSPF. The lower the cost, the higher the likelihood of using that route to forward traffic. See “Configurable metrics” on page 6-5.

Setting	Description
Autonomous System External route (ASE) type and tag	ASEs are used only when OSPF is turned off on a particular interface. When OSPF is enabled, the ASE parameters are not applicable. ASE-type specifies he type of metric that the Pipeline advertises for external routes. A Type 1 external metric is expressed in the same units as the link-state metric (the same units as interface cost). A Type 2 external metric is considered larger than any link state path. Use of Type 2 external metrics assumes that routing between autonomous systems is the major cost of routing a packet, and eliminates the need for conversion of external costs to internal link-state metrics. ASE-tag is a hexadecimal number used to tag external routes for filtering by other routers.
Transit delay	Specify the estimated number of seconds it takes to transmit a Link State Update Packet over this interface, taking into account transmission and propagation delays. On a connected route, you can leave the default of 1.
Retransmit interval	Specify the number of seconds between retransmissions of Link-State Advertisements, Database Description, and Link State Request Packets.
OSPF global option for disabling ASBR calculations	Autonomous System Border Routers (ASBRs) perform calculations related to external routes. The Pipeline imports external routes from RIP (for example, when it establishes a WAN link with a caller that does not support OSPF) and the ASBR calculations are always performed. If you must prevent the Pipeline from performing ASBR calculations, you can disable them in Ethernet>Mod Config>OSPF Global Options.

Examples of adding the Pipeline to an OSPF network

This section shows how to add a Pipeline to your OSPF network. It assumes that you know how to configure the Pipeline with an appropriate IP address. (as described in Chapter 4, “Configuring IP Routing”). The examples in this section use the network diagram in Figure 6-7 to configure the unit labeled Pipeline-1.

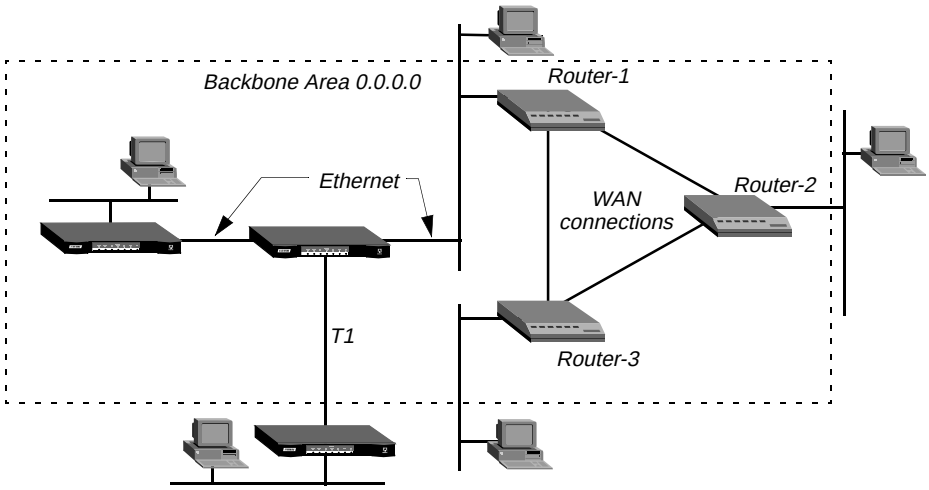


Figure 6-7. A sample OSPF setup

In Figure 6-7, all OSPF routers are in the same area (the backbone area), so the units will all form adjacencies and synchronize their databases together.

Note: All OSPF routers in Figure 6-7 have RIP turned off. OSPF can learn routes from RIP without the added overhead of running RIP.

Configuring OSPF on the Ethernet interface

The Pipeline Ethernet interface in the sample network diagram is in the OSPF backbone area. Although there is no limitation stated in the RFC about the number of routers in the backbone area, you should keep the number of routers relatively small, because changes that occur in area zero propagate throughout the AS.

Another way to configure the same units would be to create a second area (such as 0.0.0.1) in one of the existing OSPF routers, and add the Pipeline to that area. You could then assign the same area number (0.0.0.1) to all OSPF routers reached through the Pipeline across a WAN link.

After you configure the Pipeline as an IP host on that interface, you can configure it, in the Ethernet profile, as an OSPF router in the backbone area. To configure the Pipeline as an OSPF router on Ethernet, first verify that you have configured the Pipeline as an IP router.

Verifying IP connectivity

To verify IP connectivity:

1. Open the Ethernet > Mod Config > Ether1 Options (or Ether2 Options).
2. Verify that the IP Adrs setting is non-zero.

Each working Ethernet interface must be configured with a valid IP address and subnet mask. To validate the IP addresses, use another IP host on your network. Ping or Telnet to each

configured Pipeline address. If you receive no response from the Pipeline, and believe the addresses are valid for your environment, see your network administrator.

Note: It is not necessary to enable RIP and OSPF simultaneously, and disabling RIP reduces the processing overhead of the Pipeline. If you have routers on your network that support RIP only, the Pipeline uses OSPF to learn routes from RIP, incorporating them into its routing table, assigning them an external metric, and tagging them as external routes. (For more information, see Chapter 4, "Configuring IP Routing.")

Enabling OSPF routing

To enable OSPF routing on the Pipeline.

1 Open Ethernet>Mod Config>OSPF Options set RunOSPF to Yes.

```
RunOSPF=Yes
```

2 Specify the area number and area type for the Ethernet interfaces.

For example, when the Ethernet interface is in the backbone area, set Area ID to 0.0.0.0. (Because the backbone area is not a *stub* area, so you should leave the setting at its default value of 0.0.0.0. See "Stub areas" on page 6-6 for background information.)

Configuring authentication

If access to the backbone area requires authentication, specify the password:

1 In the Ethernet>Mod Config>OSPF Options, set AuthType to Simple.

If authentication is not required, set Authentication Type to None.

2 Specify the password in the AuthKey parameter.

Configuring OSPF costs and intervals

To configure metrics to identify characteristics of the route and specify how likely the Pipeline is to use this route to forward traffic. It is common practice to configure multiple routes to identical networks, and these metrics show the Pipeline how to rank each route.

1 In the Ethernet>Mod Config>OSPF Options, specify a cost greater than zero and less than 16777215. For example:

```
Cost=1
```

By default the cost of a Ethernet connected route is 1.

2 Set Transit Delay to the number of seconds it takes to transmit a Link State Update (LSU) Packet over this interface. For example:

```
TransitDelay=1
```

This value should take into account transmission and propagation delays. The default is

3 Set the retransmit interval to the number of seconds between retransmissions of OSPF packets.

```
RetransmitInterval=5
```

This specifies the number of seconds between retransmissions of Link-State Advertisements, Database Description and Link State Request Packets.

4 Exit and save the Ethernet profile.

When you upload the changes, the Pipeline comes up as an OSPF router on that interface. It forms adjacencies and builds its routing table.

Configuring OSPF across the WAN

The WAN interface of the Pipeline is a point-to-point network. A point-to-point network is any network that joins a single pair of routers. Such networks typically do not provide a broadcasting or multicasting service, so all advertisements are sent point to point.

An OSPF WAN link has a default Output Cost of 10. You can assign a higher cost to reflect a slower, lower-bandwidth connection or a lower cost to set up a preferred route to a certain destination. If the cost of one route is lower than that of another to the same destination, the higher-cost route will not be used unless route preferences change the equation.

OSPF on the WAN link is configured in a Connection profile. In this example, the Pipeline is connecting to another Pipeline unit across a nailed T1 link (as in Figure 6-7 on page 6-11).

First, configure a Connection to the remote device:

1 Open the Connection profile for the remote Pipeline unit.

2 Turn on Route IP and configure the IP routing connection. For example:

```
Ethernet
   Connections
      IP options…
         LAN Adrs=10.2.3.4/24
         WAN Alias=0.0.0.0
         IF Adrs=0.0.0.0
         Metric=7
         Preference=N/A
         Private=No
         RIP=Off
        Pool=0
```

See Chapter 4, “Configuring IP Routing.”

3 Open Connections>OSPF Options and turn on RunOSPF.

```
      OSPF options…
         RunOSPF=Yes
```

4 Specify the area number for the remote device and the area type.

The area number must always be specified in dotted-quad format similar to an IP address. For example:

```
         Area=0.0.0.0
         AreaType=Normal
         StubAreaDefaultCost=N/A
```

At this release, we recommend that you use the same area number for the Ethernet interface of the Pipeline and each of its WAN links. In this example, the Ethernet interface is in the backbone area (0.0.0.0). You can use any area numbering scheme that is consistent throughout the AS and uses this format.

5 Leave the Hello interval, Dead interval, and Priority values set to their defaults.

```
         HelloInterval=40
         DeadInterval=120
         Priority=5
```

The Priority value is used to configure the Pipeline as a DR or BDR.

6 If authentication is required to get into the backbone area, specify the password.

For example:

```
AuthType=Simple
AuthKey=ascend0
```

If authentication is not required, set AuthType=None.

7 Configure the cost for the route to Pipeline-2.

For example, for a T1 link the cost should be at least 10.

```
Cost=10
```

8 Close the Connection profile.

Of course, the remote Pipeline unit must also have a comparable Connection profile to connect to the local device.

Configuring a WAN link that doesn't support OSPF

In this example, the Pipeline has a Connection profile to a remote Pipeline unit across a BRI link (as in Figure 6-7 on page 6-11). The remote Pipeline is an IP router that uses RIP-v2 to transmit routes using RIP-v2. The route to this network, as well as any routes the Pipeline learns about from the remote Pipeline, are ASEs (external to the OSPF system).

To enable OSPF to add the RIP-v2 routes to its routing table, configure RIP-v2 normally in this Connection profile. OSPF will import all RIP routes as Type-2 ASEs.

In this example, RIP is turned off on the link and ASE information is configured explicitly.

First, configure a Connection to the remote device:

1 Open the Connection profile for the remote Pipeline unit.

2 Turn on Route IP and configure the IP routing connection. For example:

```
Ethernet
   Connections
      IP options…
         LAN Adrs=10.2.3.4/24
         WAN Alias=0.0.0.0
         IF Adrs=0.0.0.0
         Metric=7
         Preference=N/A
         Private=No
         RIP=Off
         Pool=0
```

See Chapter 4, "Configuring IP Routing." Note that Connections>OSPF Options includes two ASE parameters that are active only when OSPF is *not* running on a link. When you configure these parameters, the Connection profile route will be advertised whenever the Pipeline is up.

3 Open the OSPF Options subprofile.

4 Leave RunOSPF set to No.

```
OSPF options…
   RunOSPF=No
```

5 Configure the cost for the route to the remote Pipeline.

For example, for a single-channel BRI link could have a cost approximately 24 times the cost of a dedicated T1 link:

```
Cost=240
```

6 Specify the ASE-type metric for this route.

```
ASE-type=Type 2
```

```
ASE-type=Type 2
```

This specifies the type of metric to be advertised for an external route.

A Type 1 external metric is expressed in the same units as the link state metric (the same units as interface cost). Type 1 is the default.

A Type 2 external metric is considered larger than any link state path. Use of Type 2 external metrics assumes that routing outside the AS is the major cost of routing a packet, and eliminates the need for conversion of external costs to internal link state metrics.

7 Enter an ASE-tag for this route.

The ASE-tag is a hexadecimal number that shows up in management utilities and “flags” this route as external. It may also be used by border routers to filter this record. For example:

```
ASE-tag=cfff8000
```

8 Close the Connection profile.

Of course, the remote Pipeline unit must also have a comparable Connection profile to connect to the local device.

Setting Up IP Multicast Forwarding

7

This chapter covers the following topics:

Overview

The multicast backbone (MBONE) is a virtual network layered on top of the Internet to support IP multicast routing across point-to-point links. It is used for transmitting audio and video on the Internet in real-time, because multicasting is a much cheaper and faster way to communicate the same information to multiple hosts.

When using the MBONE, the Pipeline looks like a multicast client. It responds as a client to Internet Group Membership Protocol (IGMP) packets it receives from MBONE routers, which may be IGMP version-1 or version-2, including IGMP MTRACE (multicast trace) packets.

To multicast clients on a WAN or Ethernet interface, the Pipeline looks like a multicast router. Like a router, it sends those clients IGMP queries, receives responses, and forwards multicast traffic. In this implementation, multicast clients are not allowed to source multicast packets. If they do, the Pipeline discards the packets.

For information about monitoring and managing multicast routing, see the system administration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

Understanding the multicast parameters

This section provides some background information about multicast parameters.

(For additional details on each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.)

Enabling multicast forwarding

The Forwarding parameter turns on multicast forwarding in the Pipeline.

When you change the Forwarding parameter from No to Yes, the multicast subsystem reads the values in the Ethernet profile and initiates the forwarding function.

Note: If you modify a multicast value in the Ethernet profile, you must set this parameter to No and back to Yes again to force a read of the new value.

Specifying the MBONE interface

The MBONE interface is where the multicast router resides. If it resides across the WAN, the Mbone Profile parameter must specify the name of a resident Connection profile to that router. If the Mbone Profile name is null and Multicast Forwarding is turned on, the Pipeline assumes that its Ethernet is the MBONE interface.

Monitoring the multicast heartbeat

When it is running as a multicast forwarder, the Pipeline continually receives multicast traffic. The heartbeat-monitoring feature enables the administrator to monitor possible connectivity problems by continuously polling for this traffic and generating an SNMP alarm trap if there is a traffic breakdown. Following is a sample SNMP alarm trap:

```
Trap type: TRAP_ENTERPRISE
Code: TRAP_MULTICAST_TREE_BROKEN (19)
Arguments:
1) Multicast group address being monitored (4 bytes).
2) Source address of last heartbeat packet received (4 bytes).
3) Slot time interval configured in seconds (4 bytes).
4) Number of slots configured (4 bytes).
5) Total number of heartbeat packets received before the Pipeline
started sending SNMP Alarms (4bytes).
```

Note: Heartbeat monitoring is optional. It is not required for multicast forwarding.

To set up heartbeat monitoring, you configure several parameters that define which packets will be monitored, how often and for how long to poll for multicast packets, and

the threshold for generating an alarm. Following are the parameters you use to specify these settings:

Setting	Parameters
Which packets will be monitored	HeartBeat Addr specifies a multicast address. If specified, the Pipeline listens for packets to and from this group. HeartBeat Udp Port specifies a UDP port number. If specified, the Pipeline listens only to packets received on that port. Source Addr and Source Mask specify an IP address and netmask. If specified, the Pipeline ignores packets from that source for monitoring purposes.
How often and for how long to poll for multicast packets	HeartBeat Slot Time specifies an interval (in seconds). The Pipeline polls for multicast traffic, waits for this interval, and then polls again. HeartBeat Slot Count specifies how many times to poll before comparing the number of heartbeat packets received to the Alarm Threshold.
The threshold for generating an alarm	Alarm Threshold specifies a number. If the number of monitored packets falls below this number, the SNMP alarm trap is sent.

Configuring multicast forwarding on a client interface

Each local or WAN interface that supports multicast clients must set the Client (or Multicast Client) parameter to Yes. With this setting, the Pipeline begins handling IGMP requests and responses on the interface. It does not begin forwarding multicast traffic until the rate limit is set.

The Rate Limit parameter specifies the rate at which the Pipeline accepts multicast packets from its clients. It does not affect the MBONE interface. By default, the Rate Limit parameter is set to 100. This disables multicast forwarding on the interface. The forwarder handles IGMP packets, but does not accept packets from clients or forward multicast packets from the MBONE router.

To begin forwarding multicast traffic on the interface, you must set the Rate Limit parameter to a number less than 100. For example, if you set it to 5, the Pipeline accepts a packet from multicast clients on the interface every 5 seconds. Any subsequent packets received in that 5-second window are discarded.

An implicit priority setting for dropping multicast packets

For high-bandwidth data, voice, and audio multicast applications, the Pipeline supports both multicast rate limiting and prioritized packet dropping. If the Pipeline is the receiving device under extremely high loads, it drops packets according to a priority ranking, which is determined by the following UDP port ranges:

- Traffic on ports 0–16384 (unclassified traffic) has the lowest priority (50).
- Traffic on ports 16385–32768 (Audio traffic) has the highest priority (70).
- Traffic on ports 32769–49152 (Whiteboard traffic) has medium priority (60).
- Traffic on ports 49153–65536 (Video traffic) has low priority (55).

Forwarding from an MBONE router on Ethernet

Figure 7-1 shows a local multicast router on one of the Pipeline unit's Ethernet interfaces and multicast clients.

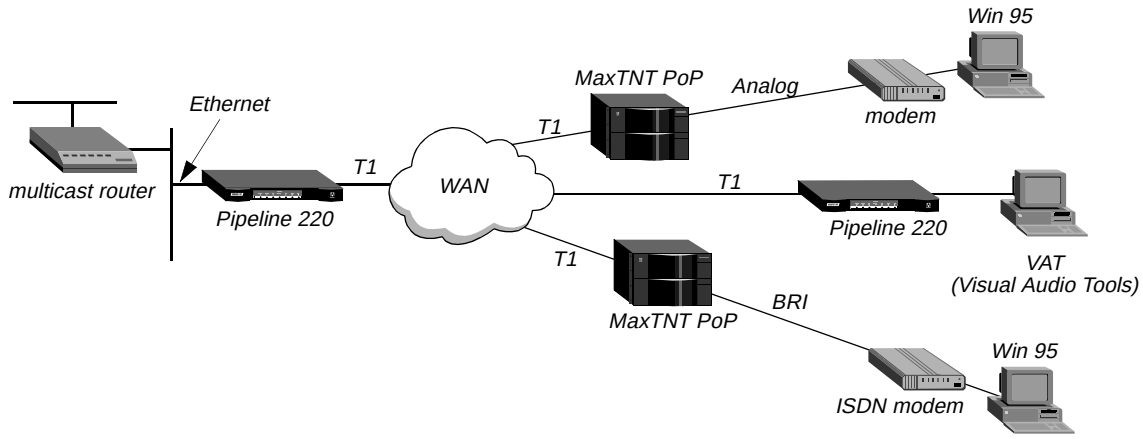


Figure 7-1. Pipeline forwarding multicast traffic to multicast clients

Note: Heartbeat monitoring is an optional feature. You can operate multicast forwarding without it if you prefer.

Configuring system-wide multicast parameters

This sample profile specifies the MBONE interface as the Ethernet port, and uses the heartbeat group address of 224.1.1.1:

1. Open Ethernet > Mod Config > Multicast.
2. Enable multicast forwarding, and leave the default values for the Mbone profile, Client, and Rate Limit parameters.

```
Ethernet
   Mod Config
      Multicast...
         Forwarding=Yes
         Mbone Profile=
         Client=No
         Rate Limit=5
```

3. Specify a heartbeat group address and UDP port for monitoring heartbeat packets.

```
HeartBeat Addr=224.1.1.1
HeartBeat Udp Port=16387
```

   To perform heartbeat monitoring, the Pipeline looks for traffic destined for this address and port.

4. Set Heartbeat Slot Time to the number of seconds between Pipeline polls for Multicast traffic. For example:

```
HeartBeat Slot Time=10
```

5 Set HeartBeat Slot Count to the quantity of polling cycles the Pipeline waits before comparing the number of heartbeat packets received to the Alarm Threshold. For example:

```
HeartBeat Slot Count=10
```

6 Set Heartbeat Alarm Threshold.

The Pipeline sends an SNMP alarm trap if the number of monitored packets falls below the Heartbeat Alarm Threshold.

7 Close the Ethernet profile.

Configuring multicasting on WAN interfaces

To enable multicasting on WAN interfaces:

1 Open the Connection profile for a multicast client site.

2 Open the IP options subprofile and set Multicast Client to Yes. If appropriate, specify a rate limit other than the default 5. Setting it to 100 disables multicast forwarding on this interface.

```
Ethernet
   Connections
      Ip options...
         Multicast Client=Yes
         Multicast Rate Limit=5
```

3 Close the Connection profile.

Forwarding from an MBONE router on a WAN link

Figure 7-2 shows a multicast router on the WAN with local and multicast clients.

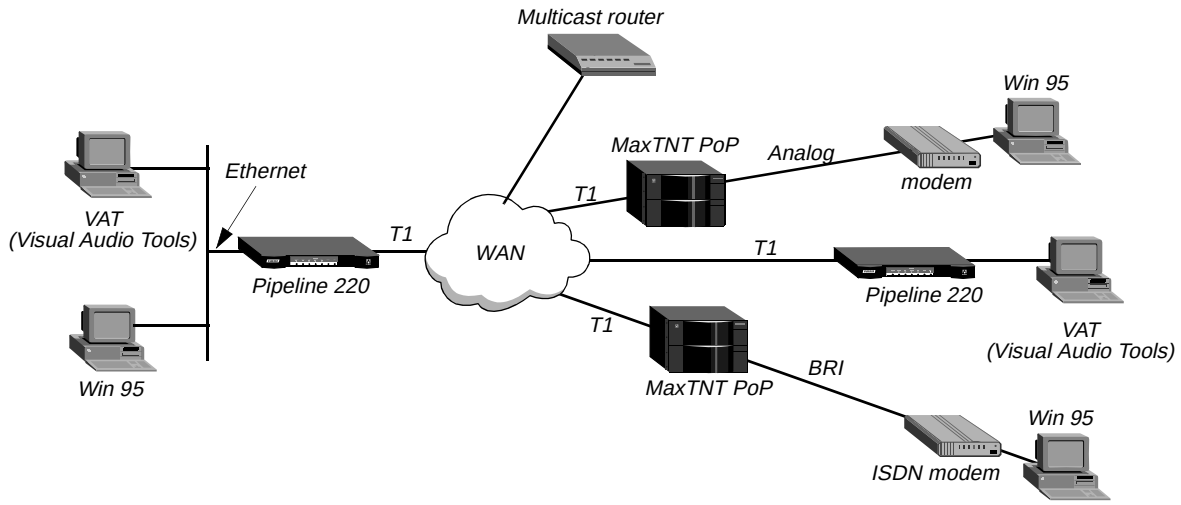


Figure 7-2. Pipeline as a multicast forwarder on Ethernet and WAN interfaces

Note: This example does not use heartbeat monitoring. If you want to configure the Pipeline for heartbeat monitoring, see the sample settings in "Forwarding from an MBONE router on Ethernet" on page 7-4.

This sample profile specifies the MBONE interface as a WAN link accessed through a Connection profile.

Configuring the Pipeline to respond to multicast clients

To configure the Pipeline to respond to multicast clients on the Ethernet:

1. Open Ethernet > Mod Config > Multicast.
2. Enable multicast forwarding, specify the number of the Connection profile for the MBONE interface, and set Client to Yes.
3. Set Multicast Rate Limit to a number lower than the default 100. Setting it to 100 disables multicast forwarding on this interface.

```
Ethernet
   Mod Config
      Multicast...
         Forwarding=Yes
         Mbone Profile=20
         Client=Yes
         Rate Limit=5
```

4. Close the Ethernet profile.

Configuring the MBONE interface

To configure the MBONE interface:

1. Open the Connection profile for a MBONE interface (in this example, profile #20).
2. Open the IP options subprofile and set Multicast Rate Limit to a number lower than the default 100. Setting it to 100 disables multicast forwarding on this interface.

```
Ethernet
   Connections
      profile #20...
         Ip options...
            Multicast Client=No
            Multicast Rate Limit=5
```

3. Close the Connection profile.

Configuring multicasting on WAN interfaces

To enable multicasting on WAN interfaces, open the connection for a multicast client site:

1. Open the Connection profile for a multicast client site.
2. Open the IP options subprofile and set Multicast Client to Yes.
3. Set Multicast Rate Limit to a number lower than the default 100. Setting it to 100 disables multicast forwarding on this interface.

```
Ethernet
   Connections
      Ip options...
         Multicast Client=Yes
         Multicast Rate Limit=5
```

4. Close the Connection profile.

Configuring IPX Routing

8

This chapter covers these topics:

Introduction to Ascend IPX routing

This section describes how the Pipeline supports IPX routing between sites that run Novell NetWare version 3.11 or newer. The Pipeline operates as an IPX router with one interface on each of its two local Ethernet interfaces and the third across the WAN. Each IPX Connection profile defines an IPX WAN interface.

The most common use for IPX routing in the Pipeline is to integrate multiple NetWare LANs to form an interconnected wide-area network

The Pipeline supports IPX routing over PPP and frame relay connections. Support for both the IPXWAN and PPP IPXCP protocols make the Pipeline fully interoperable with non-Ascend products that conform to these protocols and associated RFCs.

Note: IPX transmission can use multiple frame types. The Pipeline, however, routes only one IPX frame type (which you configure), and it routes and spoofs IPX packets only if they are encapsulated in that frame. If bridging and IPX routing are enabled in the same Connection profile, the Pipeline bridges any other IPX packet frame types. (For more information see Chapter 10, “Configuring Packet Bridging.”)

Unlike an IP routing configuration, where the Pipeline uniquely identifies the calling device by its IP address, a Pipeline IPX routing configuration does not include a built-in way to uniquely identify callers. For that reason, password authentication using PAP or CHAP is required unless IP routing is configured in the same Connection Profile.

IPX Service Advertising Protocol (SAP) tables

The Pipeline follows standard IPX SAP behavior for routers. However, when it connects to another Ascend unit configured for IPX routing, the two units exchange their entire SAP tables. Each unit immediately adds all remote services to its SAP table.

NetWare servers broadcast SAP packets every 60 seconds to make sure that routers (such as the Pipeline) know about their services. Each router builds a SAP table with an entry for each

service advertised by each known server. When a router stops receiving SAP broadcasts from a server, it ages that entry in its SAP table and eventually removes it from the table.

Routers use SAP tables to respond to client queries. When a NetWare client sends a SAP request to locate a service, the Pipeline consults its SAP table and replies with its own hardware address and the internal address of the requested server. This is analogous to proxy ARP in an IP environment.

Then, the client transmits packets whose destination address is the internal address of the server. When the Pipeline receives those packets, it consults its RIP table. If it finds an entry for that destination address, it brings up the connection or forwards the packet across the active connection.

IPX RIP (Routing Information Protocol) tables

The Pipeline follows standard IPX RIP behavior for routers when connecting to non-Ascend units. However, when two Ascend units configured for IPX routing connect, they immediately exchange their entire RIP tables. In addition, the Pipeline maintains those RIP entries as static until the unit is reset or power-cycled.

Note: In this chapter, RIP always refers to IPX RIP.

IPX RIP is similar to the routing information protocol in the TCP/IP protocol suite, but it is a different protocol.

The destination of an IPX route is the internal network of a server. For example, NetWare file servers are assigned an internal IPX network number by the network administrator and typically use the default node address of 000000000001. This is the destination network address for file read/write requests. (If you are not familiar with internal network numbers, see your NetWare documentation for details.)

IPX routers broadcast RIP updates periodically and when a WAN connection is established. The Pipeline receives RIP broadcasts from a remote device, adds 1 to the hop count of each advertised route, updates its own RIP table, and broadcasts updated RIP packets on connected networks in a split-horizon fashion.

The Pipeline recognizes network number –2 (0xFFFFFFFE) as the IPX RIP default route. When it receives a packet for an unknown destination, it forwards the packet to the IPX router advertising the default route. For example, if the Pipeline receives an IPX packet destined for network 77777777, and it does not have a RIP table entry for that destination, it forwards the packet towards network number FFFFFFFE, if available, instead of simply dropping the packet. If more than one IPX router is advertising the default route, the routing decision is based on Hop and Tick count.

Ascend extensions to standard IPX

NetWare uses dynamic routing and service location, so clients expect to be able to locate a server dynamically, regardless of where it is physically located. To help accommodate these expectations in a WAN environment, Ascend provides two IPX extensions: IPX Route profiles and IPX SAP filters.

(For information about the Handle IPX parameter and IPX bridging, see Chapter 10, “Configuring Packet Bridging.”)

IPX Route Profiles

IPX Route profiles specify static IPX routes. When the Pipeline clears its RIP and SAP tables due to a reset or power-cycle, it adds the static routes when it reinitializes. Each static route contains the information needed to reach one server.

If the Pipeline connects to another Ascend unit, some sites choose not to configure a static route. Instead, after a power-cycle or reset, the initial connection to that site, must be manually activated. After the initial connection, the Pipeline downloads the RIP table from the remote site and maintains the routes as static until the next power-cycle or reset.

Static routes need manual updating whenever the specified server is removed or has an address change. However, static routes help prevent timeouts when a client takes a long time to locate a server across a remote WAN link. (For more information, see “Configuring static IPX routes” on page 8-7, or see the *Pipeline 220 VT100 Interface Reference Guide* for information about parameters in a profile.)

IPX SAP filters

Many sites do not want the Pipeline SAP table to include long lists of all services available at a remote site. IPX SAP filters enable you to exclude services from, or explicitly include certain services in, the SAP table.

SAP filters can be applied to inbound or outbound SAP packets. Inbound filters control which services are added to the Pipeline unit’s SAP table from advertisements on a network link. Outbound filters control which services the Pipeline advertises on a particular network link. (For more information, see “Filtering SAP traffic” on page 8-9, or see the *Pipeline 220 VT100 Interface Reference Guide* for information about parameters in a profile.)

WAN considerations for NetWare client software

NetWare clients on a wide-area network do not need special configuration in most cases. These are some issues that sometimes affect NetWare clients in an IPX routing environment:

- Preferred servers

  If the local IPX network supports NetWare servers, configure NetWare clients with a preferred server on the local network, not at a remote site. If the local Ethernet does not support NetWare servers, configure local clients with a preferred server on the network with the least expensive connection costs. (See your NetWare documentation for more information.)

- Local copy of LOGIN.EXE

  Due to possible performance issues, executing programs remotely is not recommended. You should put LOGIN.EXE on each client’s local drive.

- Packet Burst (NetWare 3.11)

  Packet Burst lets servers send a data stream across the WAN before a client sends an acknowledgment. The feature is enabled by default in server and client software for NetWare 3.12 or later. If local servers are running NetWare 3.11, they should have PBURST.NLM loaded. (See your NetWare documentation for more information.)

- Macintosh or UNIX clients

  Both Macintosh and UNIX clients can use IPX to communicate with servers. But they also support native communications via AppleTalk or TCP/IP, respectively. If Macintosh

clients must use AppleTalk software (rather than MacIPX) to access NetWare servers across the WAN, the WAN link must support bridging. Otherwise, AppleTalk packets will not make it across the connection.

If UNIX clients access NetWare servers via TCP/IP (rather than UNIXWare), the Pipeline must be configured as either a bridge or an IP router. Otherwise, TCP/IP packets will not make it across the connection.

IPX in the Answer Profile

When the Pipeline receives a request to bring the WAN link up, it checks the settings in its Answer Profile. If the request does not include the information required by the Answer Profile, the Pipeline cannot successfully bring up the WAN link.

Note: Unlike an IP routing configuration, where the Pipeline uniquely identifies the calling device by its IP address, an IPX routing configuration does not include a built-in way to uniquely identify callers. For that reason, password authentication using PAP or CHAP is required unless IP routing is configured in the same Connection Profile.

Enabling IPX routing

To enable IPX routing, open the Ethernet > Mod Config profile and set IPX Routing to Yes:

```
Ethernet
   Mod Config
      IPX Routing=Yes
```

Enabling authentication

To enable password authentication for any remote request to bring up the WAN link, open the Ethernet > Answer>PPP Options submenu and select an authentication method:

```
Ethernet
   Answer
      PPP options...
         Route IPX
          Recv Auth=Either
```

Applying an IPX SAP Filter to the Answer profile

To apply an IPX SAP Filter Profile to the Answer profile:

1. Open Answer>Session Options.
2. Specify IPX SAP Filter profile by number. For example:

   ```
   Ethernet
      Answer
         Session options...
            IPX SAP Filter=1
   ```

   For details, see “Filtering SAP traffic” on page 8-9.
3. Exit and save the Answer profile.

Integrating the Pipeline into the local IPX network

To connect the Pipeline to your local IPX network, you must perform the following tasks:

- Check any local NetWare server configurations, so your Pipeline configuration is consistent.
- On the Pipeline, enable IPX routing, specify IPX frame type, and network number.

Checking local NetWare configurations

IPX packets are supported in more than one Ethernet frame type on an Ethernet segment. However, the Pipeline can only route one IPX frame type (that you specify). It will, however, bridge any other IPX packet types if bridging is enabled.

To check the IPX configuration of a NetWare server on the local Ethernet:

1 Go to the NetWare server's console.

2 Type LOAD INSTALL to view the AUTOEXEC.NCF file.

3 Look for lines similar to the following:

```
internal network 1234
Bind ipx ipx-card net=CF0123FF
Load 3c509 name=ipx-card frame=ETHERNET_8023
```

The first line specifies the internal network number of the server. If you are not familiar with internal network numbers, see your NetWare documentation. Ascend units do not require internal network numbers.

The BIND line specifies the IPX network number in use on the Ethernet. The Pipeline must use the same IPX network number for its Ethernet interface. You can specify the number explicitly in the Pipeline Ethernet Profile, or leave the Pipeline number set to zero to enable it to "learn" the number from other routers.

The LOAD line specifies the packet frame being used by this server's Ethernet controller (in this example, 802.3 frames). If you are not familiar with the concept of packet frames, see the Novell documentation.

Note: IPX network numbers on each network segment and internal network within a server on the *entire WAN* must have a unique network number. So you should know both the external and internal network numbers in use at all sites.

Configuring IPX on the Pipeline Ethernet interface

By default, when you turn on IPX routing in the Pipeline and close the Ethernet Profile, the Pipeline comes up in IPX routing mode, uses the default frame type 802.2 (which is the suggested frame type for NetWare 3.12 or later), and listens on the Ethernet to acquire its IPX network number from other IPX routers on that segment.

To configure the Ethernet profile:

1 Open Ethernet>Mod Config and set IPX Routing to Yes.

```
Ethernet
   Mod Config
      IPX Routing=Yes
```

2 Open the Ether Options subprofile.

3 Specify the frame type and set the IPX network number for the Ethernet interface. For example:

```
Ether options…
   IPX Frame=802.2
   IPX Enet #=00000000
```

Note: Make sure that the frame type you choose is consistent with the frame type in use by most servers on the local network.

4 Either configure the IPX network number of the external network to which the Pipeline is connected, or enable the Pipeline to learn its IPX network number from other IPX local routers:

– To configure the network number, make sure that the number is identical to the external network number of any other IPX router sharing the network cable with the Pipeline. If there are no other IPX routers sharing this network segment, be sure to choose a network number that is unique across the entire IPX internetwork.

– If IPX routers share the network cable with the Pipeline, you can set the network number to 00000000. This directs the Pipeline to learn it address from other local routers.

5 If more than one frame type needs to cross the WAN, make sure that Bridging is enabled. See Chapter 6, “Configuring Packet Bridging.”

```
Bridging=Yes
```

6 Exit the Ethernet profile and save the changes.

Working with the RIP and SAP tables

To manage the RIP and SAP tables, you might want to perform one or more of the following tasks:

- View the RIP and SAP tables
- Configure RIP in Connection Profiles
- Configure a static route
- Configure SAP in Connection Profiles
- Define and apply IPX SAP filters

You might also want to define standard call filters or data filters for additional control over WAN traffic and connections. (For details, see Chapter 11, “Defining Static Filters.”)

Viewing the RIP and SAP tables

For information about viewing the IPX RIP SAP tables, see the administration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

Restricting RIP in a Connection Profile

By default, the IPX RIP parameter for a connection is set to Both, indicating that RIP broadcasts will be exchanged in both directions. You can disable the exchange of RIP broadcasts across a WAN connection, or specify that the Pipeline only send or only receive RIP broadcasts on that connection. If the Pipeline does not receive RIP broadcasts from a remote unit, you should configure a static route to at least one server on that network (described completely in the next section).

To restrict RIP exchange across a WAN connection:

1. Open the Ethernet > Connections > any Connection profile > IPX options submenu.
2. Set the IPX RIP parameter to something other than its default value of Both.

   For example:

   ```
   IPX RIP=Recv
   ```

   This setting means that the Pipeline accepts RIP information from the remote IPX router but will not send its RIP information.

Configuring static IPX routes

Each static IPX route contains all of the information needed to reach one NetWare server on a remote network. When the Pipeline receives an outbound packet for that server, it finds the referenced Connection Profile and dials the connection.

You do not need to create IPX static routes to servers that are on the local Ethernet.

Most sites configure only a few IPX routes and rely on RIP for most other connections. If you have servers on both sides of the WAN connection, you should define a static route to the remote site even if your environment requires dynamic routes. If you have one static route to a remote site, it should specify a *master* NetWare server that knows about many other services. NetWare workstations can then learn about other remote services by connecting to that remote NetWare server.

Note: Remember that static IPX routes are manually administered, so they must be updated if there is a change to the remote server.

To define an IPX Route profile, proceed as in the following example:

1. Open an IPX Route profile.
2. Enter a name for the route. For example:

   ```
   Server Name=SERVER-1
   ```

3. Active the route to specify that the route should be added to the RIP table.

   ```
   Active=Yes
   ```

4. Enter the remote server's internal network number. For example:

   ```
   Network=ABC01FFF
   ```

5. Enter the remote server's node number. For example:

   ```
   Node=0000000000001
   ```

   The default 0000000000001 is typically the node number for NetWare file servers.
6. Specify the remote server's socket number. For example:

```
Socket=0451
```

Typically, Novell file servers use socket 0451.

The number you specify must be a well-known socket number. Services that use dynamic socket numbers can use a different socket each time they load, and not work with IPX Route Profiles. To bring up a connection to a remote service that uses a dynamic socket number, specify a *master* server on the remote network, that uses a well-known socket number.

7 Specify the SAP Service Type.

For example:

```
Server Type=0004
```

NetWare file servers are SAP Service type 0004.

8 Specify the distance (in hops) to the server.

For example:

```
Hop count=2
```

Usually the default of 2 is appropriate.

9 Specify the distance to the server in IBM PC clock ticks (1/18 second).

For example:

```
Tick count=12
```

Usually the default of 12 is appropriate, but you might need to increase this value for very distant servers.

10 Specify the name of the Connection profile that defines the WAN connection. For example,

```
Connection Name=TOREMOTE
```

Restricting SAP in a Connection Profile

By default, the IPX SAP parameter in a Connection Profile is set to Both, indicating that SAP broadcasts will be exchanged in both directions. If SAP is enabled to both send and receive broadcasts on the WAN interface, the Pipeline broadcasts its SAP table to the remote network and listens for service updates from that network. Eventually, both networks have a full table of all services on the WAN.

To control which services are advertised and where, you can disable the exchange of SAP broadcasts across a WAN connection, or specify that the Pipeline will only send or only receive SAP broadcasts on that connection.

To restrict SAP across a WAN connection:

1 Open a Connection profile that has IPX routing enabled.

2 Set the IPX RIP parameter to something other than its default value of Both. For example:

```
IPX SAP=Recv
```

With this setting, the Pipeline receives SAP table updates from the remote IPX router. If you do not want the Pipeline to send or receive SAP broadcasts on this connection, set IPX SAP to None

Filtering SAP traffic

IPX SAP filters include or exclude specific NetWare services from the Pipeline unit's SAP table. (Note that they do not help manage connectivity costs, unlike filters that prevent periodic RIP and SAP broadcasts from keeping a connection up unnecessarily.) IPX SAP filters control which services are added to the local SAP table or passed on in SAP response packets across IPX routing connections (*not* IPX bridging connections).

Defining an IPX SAP Filter

To define an IPX SAP filter:

1 Open an IPX SAP Filter profile.

2 Enter a name for the SAP filter. For example:

```
Name=NOSERVER-1
```

Defining an Input filter

Input filter conditions are applied to all SAP packets received by the Pipeline. They filter advertised services and exclude them from or include them in the Pipeline SAP table.

You can specify up to 12 conditions to exclude or include 12 types of services or particular services from the SAP table. The Pipeline applies these conditions in the order in which you list them: Input filter 1 followed by Input filter 2, and so forth.

1 Open Input SAP filter 01.

2 Enable the filter.

```
Valid=Yes
```

3 Set Type to Exclude.

4 Specify the service type (in hexadecimal format). For example:

```
Server Type=0004
```

Note: File servers are service type 4.

5 Specify the NetWare server's name, as configured on the server. For example:

```
Server Name=SERVER-1
```

If you want the Pipeline to include or exclude other services, repeat step 1 through step 5 for the additional service, each time opening the next input filter. Repeat again as needed.

Defining an output filter

Output filter conditions are applied to SAP response packets transmitted by the Pipeline. If the Pipeline receives a SAP request packet, it applies Output filters before transmitting the SAP response, and excludes or includes services from the response packet as specified by the filter conditions.

You can specify up to 12 conditions to exclude services from or include services in the response packets. The Pipeline applies the conditions in the order in which you list them: Output filter 1 followed by Output filter 2, and so forth.

1 Open Output SAP filter 01.

2 Enable the filter.

```
Valid=Yes
```

3 Set Type to Exclude.

4 Specify the service type (in hexadecimal format). For example:

```
Server Type=4
```

Note: File servers are service type 4.

5 Specify the NetWare server's name, as configured on the server. For example:

```
Server Name=SERVER-1
```

If you want the Pipeline to include or exclude other services, repeat step 1 through step 5 for the additional service, each time opening the next output filter. Repeat again as needed.

Applying IPX SAP filters

You can apply an IPX SAP filter to the local Ethernet, to WAN interfaces, or both.

- On the Ethernet, a SAP filter includes or excludes specific servers or services from the table. If directory services is not supported, servers or services that are not in the Pipeline table will be inaccessible to clients across the WAN.
- In the Answer profile, a SAP filter screens service advertisements from across the WAN if the remote device initiates the nailed-connection request.
- In a Connection profile, a SAP filter screens service advertisements from across the WAN if the Pipeline initiates the nailed-connection request.

(Although nailed connections do not function as switched connections, the initiation of the nailed connection is very similar to the initiation of a switched connection.)

Applying an IPX SAP filter to the Answer or Connection profile

To apply an IPX SAP Filter to the Answer profile and to a Connection profile:

1 Open Answer>Session Options.

2 Specify the IPX SAP Filter profile you want to assign to the Answer profile. For example:

```
Ethernet
   Answer
      Session options...
         IPX SAP Filter=1
```

3 Repeat the same assignment in Connections>Session Options.

```
Ethernet
   Connections
      Session options...
         IPX SAP Filter=1
```

4 Close the Connection profile.

Applying an IPX SAP filter to the Ethernet profile

To apply an IPX SAP Filter to the Ethernet profile:

1 Open Ethernet > Mod Config > Ether1 Options (or Ether2 Options).

2 Specify the IPX SAP Filter profile you want to assign to the Ethernet profile. For example:

```
IPX SAP Filter=1
```

3 Exit and Save the Ethernet profile.

A filter applied to the Ethernet interface takes effect immediately.

Example of an IPX routing connection

In this example, the Pipeline is connected to an IPX network that supports both servers and clients and will connect with a remote site that also supports both servers and clients (as shown in Figure 8-1.

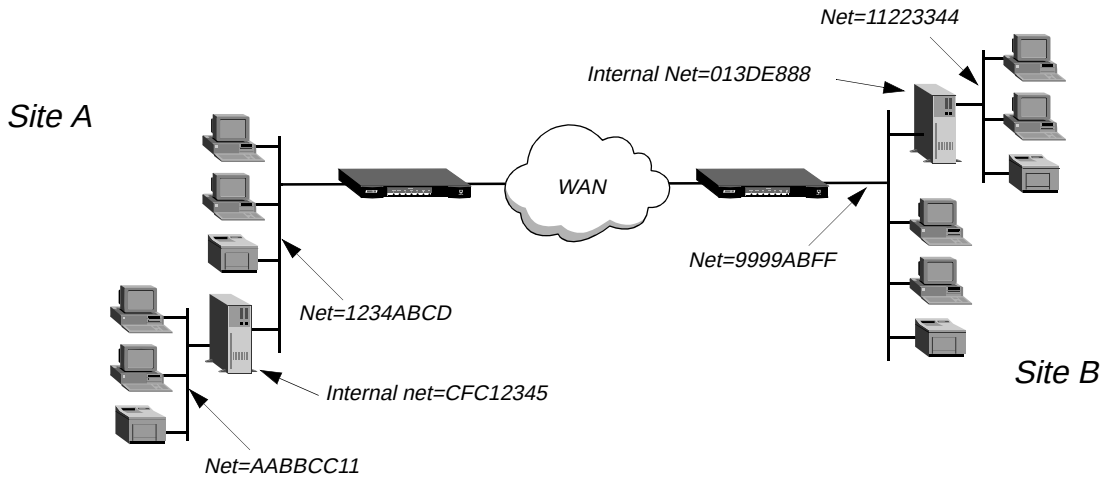


Figure 8-1. A connection with NetWare servers on both sides

Site A and site B are existing Novell LANs that support NetWare 3.12 and NetWare 4 servers, NetWare clients, and a Pipeline. The NetWare server at site A is configured with the following information:

```
Name=SERVER-1
internal net CFC12345
Load 3c509 name=ipx-card frame=ETHERNET_8023
Bind ipx ipx-card net=1234ABCD
Bind ipx ipx-card2 net=AABBCC11
```

The NetWare server at site B is configured the following information:

```
Name=SERVER-2
internal net 013DE888
Load 3c509 name=net-card frame=ETHERNET_8023
Bind ipx net-card net=9999ABFF
Bind ipx net-card2 net=11223344
```

To establish the connection shown in Figure 8-1, you would configure the Pipeline at site A, enable IPX routing for its Ethernet interface, and configure a static route to the remote server. The same procedures would apply to site B.

Configuring the Pipeline at site A

To configure the Pipeline at site A, first specify the basic information about the connection, then create a Connection profile used for connecting to site B:

1. Make sure the Pipeline has been assigned a system name in the System profile. This example uses the name SITEAGW.
2. If you haven't done so already, configure the Ethernet profile. (See "Configuring IPX on the Pipeline Ethernet interface" on page 8-5.)
3. In Answer>PPP Options, turn on IPX routing and PAP/CHAP authentication, and then close the Answer profile.

   ```
   Ethernet
      Answer
         PPP options...
            Route IPX
            Recv Auth=Either
   ```

   (If the Pipeline needs to support multiple IPX frame types, you must also enable bridging in the Answer profile.)
4. Open the Connection profile for site B.

   In this example, the Connection profile for site B is profile #5. A profile's number is the unique part of the number it is assigned in the Connections menu. For example, the Connection profile defined as 90-105 is #5.
5. Set up the Connection profile like this:

   ```
   Ethernet
      Connections
         profile 5...
            Station=SITEBGW
            Active=Yes
            Encaps=MPP
            Route IPX=Yes

            Encaps options...
               Send Auth=CHAP
               Recv PW=*SECURE*
               Send PW=*SECURE*

            IPX options...
               IPX RIP=None
               IPX SAP=Both
   ```

6. Close Connection profile #5.

Configuring a static route from site A to the remote server

Because IPX RIP is set to None in the Connection Profile, configure a static route to the remote server:

1. Open an IPX Route profile.
2. Specify that the route should be added to the RIP table:

   ```
   Active=Yes
   ```

3. Enter the remote server's internal network number. For example:

   ```
   Network=013DE888
   ```

4 Enter the remote server's node number:

```
Node=0000000000001
```

The default 0000000000001 is typically the node number for NetWare file servers.

5 Specify the remote server's socket number:

```
Socket=0451
```

Typically, Novell file servers use socket 0451.

6 Specify the SAP Service Type:

```
Server Type=0004
```

NetWare file servers are SAP Service type 0004.

7 Specify the distance (in hops) to the server:

```
Hop count=2
```

Usually the default of 2 is appropriate.

8 Specify the distance to the server in IBM PC clock ticks (1/18 second):

```
Tick count=12
```

Usually the default of 12 is appropriate, but you might need to increase this value for very distant servers.

9 Specify the name of the Connection Profile that defines the WAN connection. For example:

```
Connection #=5
```

Note: The Connection # parameter in the IPX Route profile must match the number of the Connection profile you configured to that site. The Network must specify the internal network number of the specified server.

Configuring the Pipeline at site B

To configure the Pipeline at site B, first specify the basic information about the connection:

1 Make sure the Ascend unit at site B has been assigned a system name in the System profile. This example uses the name SITEBGW.

2 Verify that the site B unit's Ethernet interface is configured for IPX routing. (See "Configuring IPX on the Pipeline Ethernet interface" on page 8-5.)

3 Verify that the site B unit's Answer profile enables IPX routing and PAP/CHAP authentication.

4 Open the Connection profile for site A.

In this example, the Connection profile for site A is profile #2. A profile's number is the unique part of the number it is assigned in the Connections menu. For example, the Connection profile defined as 90-102 is #2.

5 Set up the Connection profile like this:

```
Ethernet
   Connections
      profile 2...
         Station=SITEAGW
         Active=Yes
         Encaps=MPP
         Route IPX=Yes
```

```
Encaps options...
   Send Auth=CHAP
   Recv PW=*SECURE*
   Send PW=*SECURE*
IPX options...
   IPX RIP=None
   IPX SAP=Both
```

6 Close Connection profile #2.

Configuring a static route at site B

Because IPX RIP is set to None in the Connection Profile, configure a static route to the remote server:

1 Open an IPX Route profile.

2 Specify the name of the remote NetWare server. For example:

```
Server Name=SERVER-1
```

3 Specify that the route should be added to the RIP table.

```
Active=Yes
```

4 Enter the remote server's internal network number. For example:

```
Network=CFC12345
```

5 Enter the remote server's node number:

```
Node=0000000000001
```

The default 0000000000001 is typically the node number for NetWare file servers.

6 Specify the remote server's socket number:

```
Socket=0451
```

Typically, Novell file servers use socket 0451.

7 Specify the SAP Service Type:

```
Service Type=0004
```

NetWare file servers are SAP Service type 0004.

8 Specify the distance (in hops) to the server:

```
Hop count=2
```

Usually the default of 2 is appropriate.

9 Specify the distance to the server in IBM PC clock ticks (1/18 second):

```
Tick count=12
```

Usually the default of 12 is appropriate, but you may need to increase this value for very distant servers.

10 Specify the name of the Connection Profile that defines the WAN connection. For example:

```
Connection #=2
```

Note: The Connection # parameter in the IPX Route profile must match the number of the Connection profile you configured to that site. The Network must specify the internal network number of the specified server.

Configuring AppleTalk Routing

9

This chapter covers these topics:

Introduction to AppleTalk routing

The Pipeline functions as an AppleTalk internet router, providing routing functions for AppleTalk nodes (Macintosh workstations or Apple printers) that are connected to the Pipeline over Ethernet or a WAN. The following AppleTalk protocols are supported:

- Datagram Delivery Protocol (DDP)
- Routing Table Maintenance Protocol (RTMP)
- AppleTalk Echo Protocol (AEP)
- Zone Information Protocol (ZIP)
- Name Binding Protocol (NBP)
- AppleTalk Control Protocol (ATCP— for router-to-router applications)

When to use AppleTalk routing

With AppleTalk routing, connect two or more networks that have AppleTalk nodes, such as Mac OS computers or Apple printers. The primary benefits of routing AppleTalk traffic (as opposed to bridging this traffic) are:

- reducing broadcast and multicast traffic over the WAN
- providing startup information to local AppleTalk devices

Reducing broadcast and multicast traffic

Because AppleTalk uses multicast and broadcast addresses extensively, routing AppleTalk can greatly improve the efficiency of a LAN or WAN. By using AppleTalk zones to segment traffic, you can significantly reduce the amount of broadcast and multicast traffic on a LAN or WAN. When you set up a router for the first time, you identify the cable range (network number) for the subnetwork segment and one or more zones.

For example, when a user on a network without a router selects a device in the Chooser, the MAC OS computer sends out an Name Binding Protocol (NBP) Lookup as a broadcast packet. Since a bridge forwards all broadcast traffic, all devices on the network receive the Lookup

packet. A router can significantly reduce AppleTalk traffic over the WAN because it does not forward broadcast traffic from one subnetwork to another, but stops it at the subnetwork port of the router.

Zone multicasting is intended to prevent any node not in the destination zone for the lookup from receiving the lookup packet. Any AppleTalk node responds only to NBP lookups for that node's zone name. In the example above, a router would convert the Broadcast Request packet generated by the Lookup request to a Forward Request packet for each network that contains nodes in the target zone specified by the Lookup request.

A bridge can filter directed traffic between two specific nodes but cannot filter broadcast or multicast traffic, since there isn't a specific port that can be assigned to a multicast or broadcast address. This means that although filters used with bridging can reduce the number of AppleTalk packets sent to remote network segments, bridging does not reduce the number of broadcast and multicast packets over these networks.

Providing dynamic startup information to local devices

In addition to routing services, the Ascend AppleTalk router provides startup information to AppleTalk stations. Like other routed protocols, AppleTalk station or *node* addresses are comprised of a unique network number/node combination. AppleTalk addresses are dynamically assigned when a node starts up. In addition, the router provides an AppleTalk node with the network cable range to which it is attached and supplies zone name information.

Understanding AppleTalk zones and network ranges

AppleTalk zones and network ranges are configured in AppleTalk routers. Network numbers are assigned to network segments, and must be unique within the internetwork. A network range is a range of network numbers set into the port descriptor of the router port and then transmitted through RTMP to the other nodes of the network. Each of the numbers within a network range can represent up to 253 devices.

AppleTalk zones

A zone is a multicast address containing an arbitrary subset of the AppleTalk nodes in an internet. Each node belongs to only one zone, but a particular extended network can contain nodes belonging to any number of zones. Zones provide departmental or other groupings of network entities that a user can easily understand.

In the Ascend AppleTalk router, zone names are case-insensitive. However, since some routers regard zone names as case-sensitive it is advisable to be consistent in spelling zone names when you configure multiple connections or routers.

Extended and non-extended AppleTalk networks

There are two types of subnetworks: non-extended and extended. Non-extended networks theoretically allow up to 254 nodes. A non-extended network has one network number (not a range) and one zone. Examples of a non-extended network are LocalTalk and ARA dial-up networks.

An extended network is a group of non-extended networks on the same physical data link, and contains a range of network numbers, with each network in the range supporting up to 253 devices. EtherTalk and TokenTalk are examples of extended networks.

At least one router on a network, called the seed router, must have the network number range set into its port description. Other routers on the network can have a network range of 0, since they acquire the network number range from RTMP packets sent by the seed router. AppleTalk routers on a network must not have conflicting port network number ranges for that network. A 0 value does not cause a conflict, but otherwise, all seed routers on the same network must have the same value for the start and end of the network number range.

Figure 9-1 shows a network with three routers and three zones configured. Each zone has a range of network numbers.

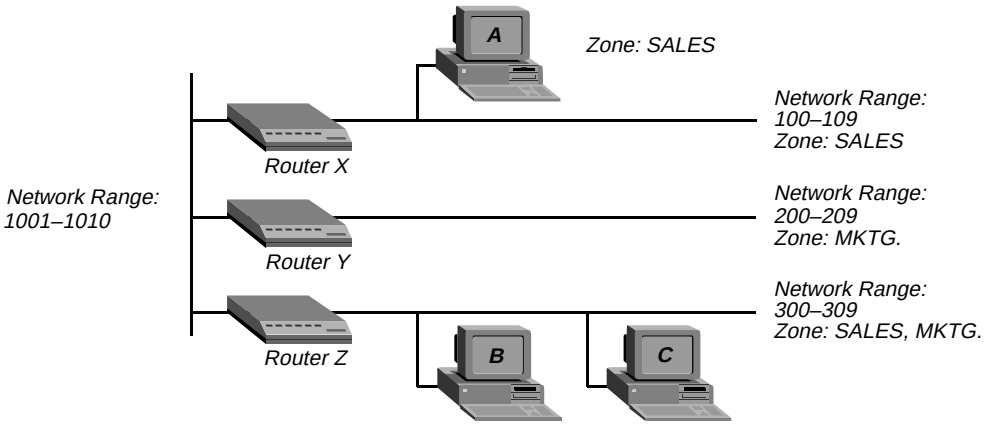


Figure 9-1. AppleTalk LAN

Router X, Router Y, and Router Z connect to the backbone network (Range 1001-1010). Each router has an additional connection to a local network segment. For example, Router X has a connection to the network range 100-109. User A's computer also connects to the 100-109 range.

Because Router X is configured with only one zone, any AppleTalk device joining the segment belongs to the SALES zone. But User B's computer can belong to either the SALES zone or the MKTG. zone. Some AppleTalk devices allow you to select the zone to which they belong. If there is no way to manually assign the zone, the AppleTalk device is put into the *default* zone, which is defined on the AppleTalk router.

Figure 9-1 shows two important concepts about network numbers and zones. When a network range is defined, all values within that range are unusable for any other segment. The segment to which user C's computer connects uses network range 300-309. No other network segment in this AppleTalk network can use network numbers 300, 301, 302, etc. in their ranges. As an example, network number 310 *is* available to a new network segment

Zones can be shared among network segments. In Figure 9-1, network 100-109 supports zone SALES. So does network 300-309.

How AppleTalk works

The following is a brief description of how the workstation user sees a typical AppleTalk connection and describes in a general way what is happening as the user makes the choices that lead to a connection. This example supposes a connection between a workstation on a Pipeline connected over to another Pipeline over Ethernet on a synchronous PPP connection, shown in Figure 2.

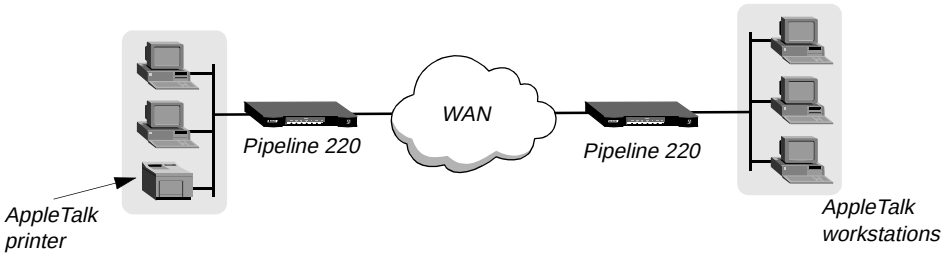


Figure 9-2. Routed connection

1. An AppleTalk workstation user opens the Chooser for the first time since it has been attached to the router and configured.

   The zones that appear are the local Ethernet zone (in this case the WAN zone is the same as the local Ethernet zone), configured in the Connection Profile for the Pipeline. This information is stored in the Pipeline.

2. The workstation sends a ZIP Query to obtain an updated zone list from the Pipeline, and the Pipeline returns the updated zone list. This list may contain different zones from the list that initially appears.

3. The user selects a zone and a specific device in the Chooser.

4. The workstation sends an NBP Broadcast Request to the Pipeline, which checks its Zone Information Table to determine in which subnetwork that printer is located, and sends the request to the Pipeline via the port configured in the Connection Profile.

5. The Pipeline determines the port to which the subnetwork is attached and performs the lookup in the appropriate multicast address (multicast addresses are assigned to zones).

6. All devices in the appropriate zone on the subnetwork hear and pick up the NBP Lookup.

7. The selected printer obtains the sender's address from the Lookup packet (in this case the routers are *forwarders*; the workstation is the *sender*) and sends the reply through the routers to the workstation.

8. The user sends the print job to the printer.

9. When the print job is complete and no data packets are passing through the connection, the Pipelines continue to pass routing information.

Configuring AppleTalk routing

To configure AppleTalk routing, you must complete the steps outlined in "System-level AppleTalk routing parameters" and "Per-connection AppleTalk routing parameters" (if required).

System-level AppleTalk routing parameters

To configure system-level AppleTalk parameters:

1. Open the Ethernet > Mod Config profile.
2. Enable AppleTalk by setting the AppleTalk parameter to Yes.

   ```
   AppleTalk=Yes
   ```

3. Open the AppleTalk submenu and set Zone Name to the zone to which the Pipeline is located. For example:

   ```
   Zone Name=SLC Finance
   ```

4. Using the AppleTalk Router parameter, specify whether the router is a seed or non-seed router.

   A seed router has a manually defined network configuration. When a non-seed router boots, it has no local network configuration. It examines local network traffic and learns its local network configuration.

   Note: You should configuring the Pipeline as a non-seed router provided there is *at least one* seed router on the local network. Having only one seed router on a local network simplifies potential network configuration changes. Should you need to change the network numbering, only the seed router needs to be reconfigured. The remaining non-seed routers simply need to be rebooted to learn the changes.

5. If the Ascend unit is to be a seed router, specify the Net Start and Net End.

   If there are other seed routers sharing the Pipeline's network segment, this information must be identical on *all* routers that *share the network segment*. If there are no other seed routers, every network number from Network Start to Network End must be unique for the entire internet. Valid network numbers are of from 1 to 65,534.

6. Set Default Zone and any other zones assigned to the local AppleTalk network segment.

   The Default Zone is assigned to any AppleTalk device that is connected to the Pipeline's local Ethernet segment that has not explicitly been assigned to another zone.

   The Default Zone and additional zone list need to be identical for any AppleTalk router sharing the local network segment.

   Note: Zones can be shared across network segments.

7. Exit and save the Ethernet > Mod Config profile.

Per-connection AppleTalk routing parameters

To configure per-connection AppleTalk parameters:

1. Open a Connection profile.
2. Set Encaps to PPP.

   The Pipeline supports AppleTalk routing over PPP-encapsulated links.

3 Set Route AppleTalk to Yes.

4 Open the AppleTalk Options submenu.

5 Set Zone Name to the zone to which the remote AppleTalk router belongs.For example:

```
Zone Name=Marketing
```

6 Set Net Start and Net End to the network range to which the remote AppleTalk router belongs.

7 Exit and save the Connection profile.

Configuring Packet Bridging

10

This chapter covers the following topics:

Introduction to Ascend bridging

This section provides an overview of packet bridging and explains how the Pipeline brings up a bridging connection.

The Pipeline is used as a bridge primarily to provide connectivity for protocols other than IP, IPX, and AppleTalk although it can also be used for joining segments of an IP, IPX, or AppleTalk network. Because a bridging connection forwards packets at the hardware-address level (link layer), it does not distinguish between protocol types, and it requires no protocol-specific network configuration.

The most common uses of bridging in the Pipeline are:

- To provide any non-routed protocol connectivity with another site
- To link any two sites so that their nodes appear to be on the same LAN
- To support protocols that depend on broadcasts to function, such as BOOTP

For information about monitoring and managing bridging on the Pipeline, see the system administration chapter in the *Pipeline 220 VT100 Interface Reference Guide*.

Disadvantages of bridging

Bridges examine *all* packets on the LAN (termed *promiscuous mode*) so they incur greater processor and memory overhead than routers. On heavily loaded networks, this increased overhead can result in slower performance.

Routers have other advantages over bridging. Because they examine packets at the network layer (instead of the link layer), you can filter on logical addresses, providing enhanced security and control. In addition, routers support multiple transmission paths to a given destination, enhancing the reliability and performance of packet delivery.

How a bridged WAN connection is initiated

When the Pipeline is configured for bridging, it accepts all packets on the Ethernet and forwards only those that have one of the following:

- A physical address that is not on the local Ethernet segment (the segment to which the Pipeline is connected).
- A broadcast address.

The important thing to remember about bridging connections is that they operate on physical and broadcast addresses, not on logical (network) addresses.

Physical addresses and the bridge table

A physical address is a unique hardware-level address associated with a specific network controller. A device's physical address is also called its Media Access Control (MAC) address. On Ethernet, the physical address is a six-byte hexadecimal number assigned by the Ethernet hardware manufacturer, for example:

```
0000D801CFF2
```

If the Pipeline receives a packet whose destination MAC address is not on the local network, it first checks its internal bridge table (for a description of the table, see "Transparent bridging" on page 10-4). If it finds the packet's destination MAC address in its bridge table, the Pipeline dials the connection and bridges the packet.

If the address is *not* specified in its bridge table, the Pipeline checks for active sessions that have bridging enabled. If there are one or more active bridging links, the Pipeline forwards the packet across *all* active sessions that have bridging enabled.

Broadcast addresses

A broadcast address is recognized by multiple nodes on a network. For example, the Ethernet broadcast address at the physical level is:

```
FFFFFFFFFFFF
```

All devices on the same network receive all packets with that destination address. When configured as a router only, the Pipeline discards broadcast packets. When configured as a bridge, it forwards packets with the broadcast destination address across all active sessions that have bridging enabled.

ARP broadcast packets that contain an IP address specified in the bridge table are a special case. For details, see "Static bridge table entries" on page 10-5.

How the Pipeline establishes a bridging connection

The Pipeline uses station names and passwords to sync up a bridging connection, as shown in Figure 10-1.

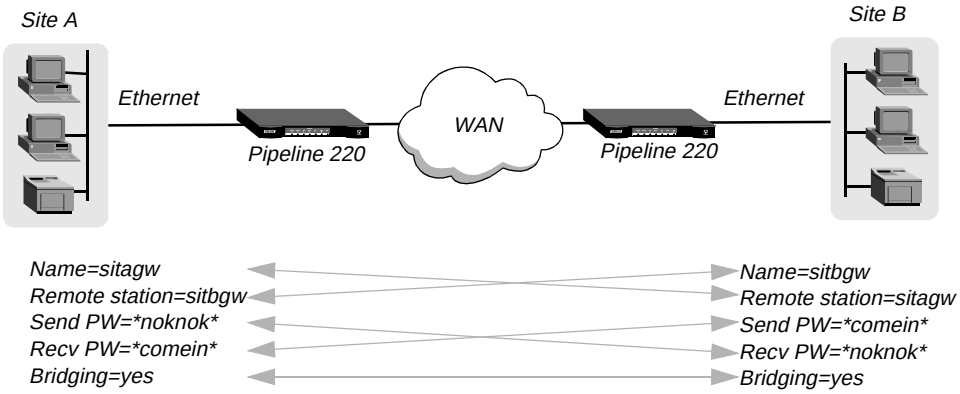


Figure 10-1. Negotiating a bridge connection (PPP encapsulation)

The system name assigned to the Pipeline in the Name parameter of System>Sys Config must *exactly* match the device name specified in the Connection profile on the remote bridge, including case changes. Similarly, the name assigned to the remote bridge must exactly match the name specified in the Station parameter of that Connection profile, including case changes.

Note: The most common cause of trouble when initially setting up a PPP bridging connection is that the wrong name is specified for the Pipeline or the remote device. Often case changes are not specified, or a dash, space, or underscore is not entered.

Enabling bridging

The Pipeline has a system-wide bridging parameter that must be enabled for any bridging connection to work. The Bridging parameter directs the Pipeline unit's Ethernet controller to run in promiscuous mode. In promiscuous mode, the Ethernet driver accepts all packets, regardless of address or packet type, and passes them up the protocol stack for a higher-layer decision on whether to route, bridge, or reject the packets. (Even if no packets are actually bridged, running in promiscuous mode incurs greater processor and memory overhead than the standard mode of operation for the Ethernet controller. On heavily loaded networks, this increased overhead can result in slower performance).

To enable bridging on the Ethernet interface open Ethernet>Mod Config and set the Bridging parameter to Yes:

```
Ethernet
   Mod Config
      Bridging=Yes
```

Bridging in the Answer Profile

Bridging must be enabled on both the local and remote side of a PPP-encapsulated link. Otherwise the link cannot bridge packets. In addition, PAP or CHAP authentication is required for unique identification of devices.

For details about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.

Note: Unlike an IP routing configuration, where the Pipeline uniquely identifies the remote device by its IP address, a bridging configuration does not include a built-in way to identify incoming callers. For that reason, password authentication using PAP or CHAP is required unless IP routing is configured in the same Connection Profile.

To set Answer Profile parameters for a bridging connection open the Ethernet > Answer > PPP options submenu and set Bridge to Yes:

```
Ethernet
   Answer
      PPP options...
         Bridge=Yes
```

Managing the bridge table

To forward bridged packets to the correct destination network, the Pipeline uses a bridge table that associates end nodes with particular connections. It builds this table dynamically, as described in “Transparent bridging” on page 10-4. It also incorporates the entries found in its Bridge profiles. Bridge profiles are analogous to static routes in a routing environment. You can define up to 99 destination nodes and their connection information in Bridge profiles.

Transparent bridging

The Pipeline is a transparent bridge (also termed a *learning bridge*). It keeps track of where a particular address is located, and of the Connection profile that specifies the interface to which the packet should be forwarded. As it forwards a packet, the Pipeline logs the packet’s source address and creates a bridge table that associates node addresses with a particular interface.

For example, Figure 10-2 shows the physical addresses of some nodes on the local Ethernet and at a remote site. The Pipeline at site A is configured as a bridge.

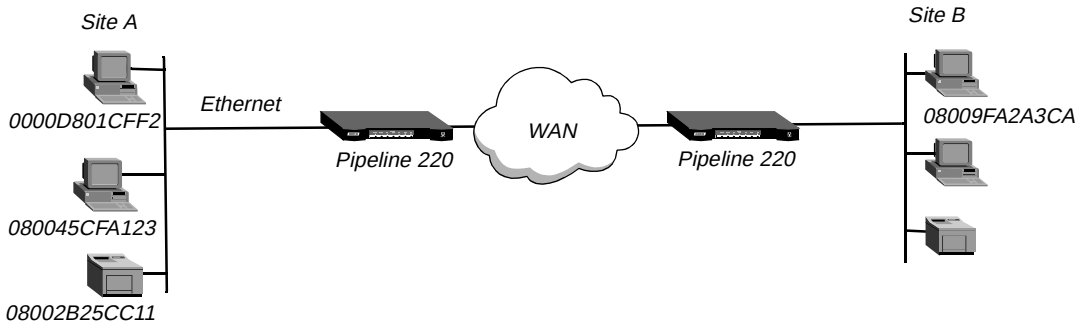


Figure 10-2. How the Pipeline creates a bridging table

The Pipeline at site A gradually *learns* addresses on both networks by looking at each packet's source address, and it develops a bridge table like this:

```
0000D801CFF2       SITEA
080045CFA123       SITEA
08002B25CC11       SITEA
08009FA2A3CA         SITEB
```

Entries in the Pipeline unit's bridge table must be relearned within a fixed aging time limit, or they are removed from the table.

Static bridge table entries

You can specify up to 99 static bridge table entries for the Pipeline.

To define a static bridge table entry:

1. Open the Ethernet > Bridge Adrs profile.
2. Enter the physical (MAC) address of the remote host. For example:

   ```
   Enet Adrs=0080AD12CF9B
   ```

   For a description of physical addresses, see "Physical addresses and the bridge table" on page 10-2 if you need more details. You must get this address from the administrator of the far-end device.
3. If the far-end is a segment of the local IP network, specify an address on that segment.

   For example:

   ```
   Net Adrs=10.1.2.133
   ```
4. Exit and Save the Bridge Adrs profile.

An example of a bridged connection

This section explains how to configure bridging for a Pipeline connecting to a remote site. The sample configuration does not show the link-specific settings, or additional routing settings that might be appropriate at your site. It focuses only on bridging. (For details about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*.)

An example bridged connection

In the following example, two Pipeline units are configured as bridges. A bridged connection at the link layer requires a bridge at both ends of the connection. (The most common cause of trouble when initially setting up a bridging connection is that the wrong name is specified for the Pipeline or the remote device. Often case changes are not specified correctly, or a dash or underscore is entered incorrectly. Make sure you type the name exactly as it appears in the remote device.)

This example assumes that you have enabled bridging for the system as described in "Enabling bridging" on page 10-3. It also assumes you have enabled bridging for incoming sessions, as described in "Bridging in the Answer Profile" on page 10-4

In this example, two segments of an IP network are connected across the WAN, as shown in Figure 10-3.

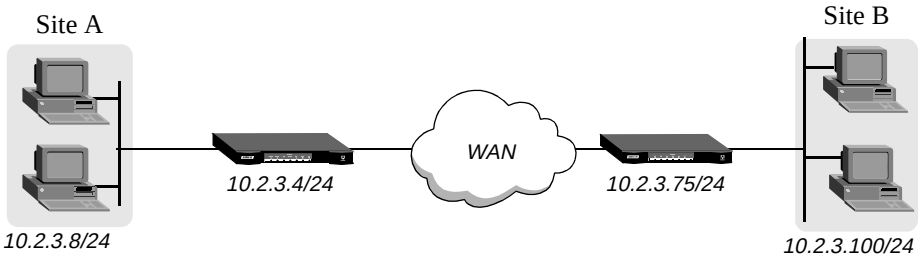


Figure 10-3. An example bridging connection

Configuring the Site A Pipeline for a bridged connection consists of assigning a system name to the Pipeline unit, configuring a bridging connection, and defining a static bridge-table entry. The settings have to be saved after configuring the connection, and again after defining the table entry, because they do not take effect until the Pipeline uploads them.

To avoid unnecessary traffic across the WAN, you can configure the Pipeline to reply to ARP requests for any remote device for which it has a bridge-table entry.

To configure the site A Pipeline for a bridged connection, assign a name to the Pipeline, configure and save the bridging Connection profile, and configure and save a static bridge-table entry.

Assigning a name

To assign a name for the site A Pipeline, Open the System>Sys Config menu and set the Name parameter.

Bridged connections use the Name for authentication.

Configuring a bridging connection

To configure a bridging connection:

1 Turn on bridging and specify an authentication protocol in Ethernet>Answer>PPP Options.

```
Ethernet
   Answer
      PPP options...
         Bridge=Yes
         Recv Auth=Either
```

2 Open Connection profile #5 and set these parameters:

```
Ethernet
   Connections
      profile #5...
         Station=SITEBGW
         Active=Yes
         Encaps=PPP
          Bridge=Yes
```

3 Configure password authentication.

```
Encaps options...
   Send Auth=CHAP
   Recv PW=localpw
   Send PW=remotepw
```

– Set Send PW to the password the Pipeline sends to the remote device to authenticate itself.

– Set Recv PW to the password the Pipeline expects from the remote device.

4 Close Connection profile #5 and save your changes.

Defining a static bridge-table entry

To define a static bridge table entry:

1 Open Ethernet>Bridge Adrs.

2 Specify a node's Ethernet address on the remote network, and the number of the Connection profile to establish a link to that network.

```
Ethernet
   Bridge Adrs
      Enet Adrs=0080AD12CF9B
      Net Adrs=0.0.0.0
      Connection #=5
```

For a description of physical addresses, see "Physical addresses and the bridge table" on page 10-2. You must get this address from the administrator of the far-end device.

3 If the far-end is a segment of the local IP network, specify an address on that segment.

For example:

```
Net Address=10.2.3.100
```

4 Close the Bridge profile.

Configuring proxy mode on the Pipeline

If you are bridging between two segments of the same IP network, you can use the Net Adrs parameter in a Bridge profile to enable the Pipeline to respond to ARP requests while bringing up the bridged connection.

If an ARP packet contains an IP address that matches the Net Adrs parameter of a Bridge Profile, the Pipeline responds to the ARP request with the Ethernet (physical) address specified in the Bridge Profile and brings up the specified connection. In effect, the Pipeline acts as a proxy for the node that actually has that address.

Defining Static Filters

11

This chapter covers the following topics:

Introduction to Ascend filters

Ascend filters define packet conditions. When a filter is in use, the Pipeline examines every packet in the packet stream and takes action if the defined filter conditions are present. The action the Pipeline takes depends both on the conditions specified within the filter and how the filter is applied.

If you are using IP Security or Secure Access Firewalls, see the Secure Access Manager (SAM) documentation for information on configuring IP Security and firewalls.

How conditions are specified

Without filtering, the Pipeline forwards all packets. The conditions specified within a filter can specify not to forward certain packets, or not to forward *any* packets *except* those defined in the filter. The conditions also specify whether the Pipeline will examine inbound packets, outbound packets, or both. (For a more detailed discussion of specifying conditions, see “Overview of Filter profiles” on page 11-3.

A filter’s forwarding action affects the actual data stream. Certain packets are dropped or forwarded, as specified in the filter conditions. Filters are often used for network security purposes, but they can be used for any purpose that requires the Pipeline to drop or forward only specific packets. For example, you can use filters to drop packets addressed to particular hosts or to prevent broadcasts from going across the WAN. On the other hand, you can use filters to allow only specific devices to be accessed by users across the WAN.

Note: With Ascend units that accept switched calls, filters can be applied as *call* filters. If a filter is applied as a call filter, its forwarding action does not affect which packets are sent across an active connection. In a call filter, the *forward* action determines which packets can either initiate a connection or reset the timer for an established connection. Call filters are typically used to prevent unnecessary connections. Because the Pipeline supports nailed WAN connections only, call filters are not necessary.

Applying a filter to the Answer profile

A filter applied to the Answer profile takes effect only when the connection goes from an offline state to an active state.

Filters applied in the Answer profile are not used if the WAN link uses a Connection profile. When a link is brought up, any configured filter in the Ethernet > Answer > Session Options > Data Filter parameter is not used. If no filter is configured in the Connection profile, then no filter is applied.

After you have *defined* a filter, you can *apply* it to the Answer profile as follows:

1. Open the Ethernet>Answer >Session Options.
2. Specify the filter's number in the Data Filter parameter. For example:

   ```
   Data Filter=1
   ```

   (Call filters are not applicable to the local network interface.)
3. Close the Ethernet profile.

   No filter is applied if Data Filter is set to 0 (the default).

Applying a filter to a Connection profile

After you have defined a filter, you can apply it to a specific Connection profile as follows to specify which packets will be allowed to cross a WAN interface:

1. Open the Session Options subprofile of the Connection profile you want to apply the filter to.
2. Specify the filter's number in the Data Filter parameter. For example:

   ```
   Data Filter=5
   ```

   Specify the unique portion of the number preceding the filter's name in the Filters menu.
3. Close the Connection profile.

Applying a filter to the Ethernet interface

After you have defined a filter, you can apply it as follows to specify which packets will be allowed to cross Ethernet interface:

1. Open the Ethernet>Mod Config>Ether Options.
2. Specify the filter's number in the Filter parameter. For example:

   ```
   Ethernet
      Mod Config
         Ether options...
            Filter=1
   ```

   (Call filters are not applicable to the local network interface.)
3. Close the Ethernet profile.

A filter applied to the Ethernet interface takes effect immediately. If you change the Filter Profile definition, the new filters are applied as soon as you save the Filter Profile, uploading the changes to the Pipeline.

Overview of Filter profiles

The three basic types of filters are Generic filters, IP filters, and IPX filters. Generic filters examine the byte- or bit-level contents of packets. They focus on bytes or bits at particular locations, and compare the contents of a location with a value defined in the filters. Protocol specifications are usually the best source of the information you need for effective use of Generic filters.

IP filters examine higher-level fields specific to IP, TCP, and UDP packets. IP filters focus on known fields in IP packets, such as source or destination address, protocol number, and so forth. They operate on logical information, which is relatively easy to obtain.

Figure 11-1 shows how filters are organized and the terminology used to describe each part of a filter.

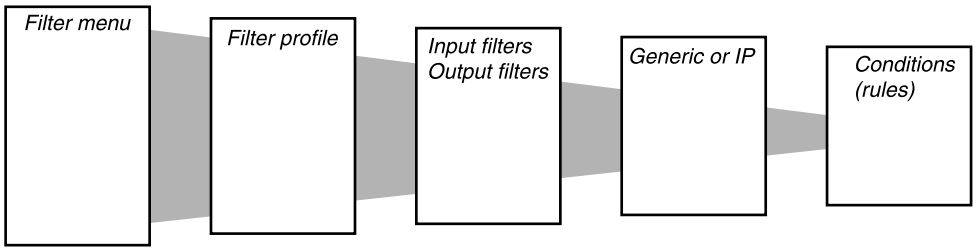


Figure 11-1. Filter terminology

The menus shown in Figure 11-1 are nested submenus below the Filters menu. The following table describes the structure of Ascend filters:

Menu	Description
Filters menu	The Filters menu displays the names of configured filers (if any). When applying a filter, you identify it by its number.
Filter profile	Each Filter profile contains a set of filter conditions for a particular filter
Input or Output filters	At the top level of a Filter profile are submenus labeled Input Filters and Output Filters. Each submenu contains a list of 12 filters. The conditions you define within those *Input Filters* or *Output Filters* (or both) are applied to the inbound or outbound packet stream, respectively, in the order in which they appear (1–12). For details, see “Filtering inbound and outbound packets” on page 11-4.
Generic, IP, or IPX filters	Each *Input Filter* and *Output Filter* can be of type GENERIC, IP, or IPX. Once you assign a type, you can open the corresponding submenu to define the packet-level filter conditions. For details, see “Selecting filter type and activating the filter” on page 11-4.
Filter conditions	Filter conditions specify the actual packet characteristics that will be examined in the data stream. Generic filters conditions specify locations and values that can be found within any packet. IP filter conditions specify IP-specific packet characteristics, such as address, mask, and port.

Filtering inbound and outbound packets

At the top level of a Filter profile, you assign a name and select either the Input Filters or Output Filters submenu.

Input filters cause the Pipeline to compare against received packets. If the filter is applied to the Ethernet interface, any Input filters are compared to packets from the Ethernet *into* the Pipeline. If applied as a filter on a WAN interface (applied to a Connection Profile), it affects packets from that WAN interface *into* the Pipeline.

Output filters cause the Pipeline to compare against packets to be sent. If the filter is applied to the Ethernet interface, any Output filters are compared to packets in the Pipeline destined for the Ethernet interface. If applied as a filter on a WAN interface (applied to a Connection Profile), it affects packets in the Pipeline destined for the WAN interface.

You can specify up to 12 Input Filters and 12 Output Filters in a Filter profile. They are applied in order from 1 to 12.

For security reasons, if a packet does not match *any* of the defined conditions, it is discarded.

However, if *only* Input filters are defined, the default action for *Output* filters is to forward all packets. The same is true in the other direction. If you define only Output filters, the default action for inbound packets is to forward them.

Selecting filter type and activating the filter

The Input Filters and Output Filters you define are applied to a packet in numerical order from one to twelve, provided that each filter has the Valid parameter set to Yes. Setting the Valid parameter to No in a filter prevents it from being applied.

After you open an Input Filter or an Output Filter, set Valid to Yes, and specify the type of filter conditions to be defined (Generic, IP, or IPX).

Generic filter conditions define bits and bytes within a packet. They are applied to all packet types. IP filter conditions apply only to TCP, IP, and UDP packets. IPX filter conditions apply only to IPX packets.

Defining generic filter conditions

If you specify a Generic filter, you define conditions for a Generic filter. To define a Generic filter, you set the following parameters (for complete information about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide).*

Parameter	Description
Forward	Determines whether the Pipeline will forward a packet if it matches the definition or discard the packet if it matches.
Offset, Length, Mask, and Value	You use the Offset, Length, Mask, and Value parameters to define the exact location of certain bytes within a packet and the values of those bytes.

Parameter	Description
Comparison	Specifies how a packet's contents are compared to the value specified in this filter. After applying the Offset, Mask, and Length values to reach the appropriate location in a packet, the contents of that location are compared to the Value parameter. If Compare is set to `Equals` (the default), the filter is applied if the packet data are identical to the specified value. If Comparison is set to `NotEquals`, the filter is applied if the packet data are not identical.
Link this condition to the next...	Specifies whether the current filter is linked to the one immediately following it. If you select this checkbox, the filter can examine multiple non-contiguous bytes within a packet by *linking* the current filter to the next one, so that the next filter is applied before the Filter Action decision is made. The match occurs only if *both* non-contiguous bytes contain the specified values. If the check box is cleared, the Filter Action decision is based on whether the packet matches the definition in this one filter.

Defining IP filter conditions

If you specify an IP filter, you define conditions related only to TCP, IP, and UDP packet filtering. These packets are compared regardless of whether the Pipeline routes or bridges IP. An IP filter examines source addresses, destination addresses, IP protocol type and port, or a combination of these.

To define an IP filter, you set the following parameters (for complete information about each parameter, see the *Pipeline 220 VT100 Interface Reference Guide*).

Parameter	Description
Forward	Determines whether the Pipeline will forward a packet if it matches the definition or discard the packet if it matches.
Source and destination address and mask	Specify the contents of the source or destination fields in a packet. Use the address parameter to specify the source or destination address, and the Mask parameter to mask out portions of the address (for example, to mask out the host number).

Parameter	Description
Protocol	Identifies a specific TCP/IP protocol (for example, 6 specifies TCP packets). Common protocols are listed below, but protocol numbers are not limited to this list. For a complete list, see the section on Well-Known Port Numbers in RFC 1700, *Assigned Numbers*, by Reynolds, J. and Postel, J., October 1994. 1 — ICMP 5 — STREAM 8 — EGP 6 — TCP 9 — Any private interior gateway protocol (such as IGRP) 11 — Network Voice Protocol 17 — UDP 20 — Host Monitoring Protocol 22 — XNS IDP 27 — Reliable Data Protocol 28 — Internet Reliable Transport Protocol 29 — ISO Transport Protocol Class 4 30 — Bulk Data Transfer Protocol 61 — Any Host Internal Protocol 89 — OSPF
Source and destination Port Compare and Port #	Specify whether to compare the protocol ports, which identify the application running over TCP/IP. The comparison can match a protocol port number that is less-than, greater-than, equal, or not-equal.
TCP Estab	Set this parameter to compare a packet only if is part of a TCP session that is already established.

Examples of filters

This section provides step-by-step examples that show how to specify Generic, IP, and IPX filter conditions.

An example generic filter to handle AppleTalk broadcasts

This section shows how to define a Generic filter whose purpose is to prevent local AppleTalk AEP and NBP traffic from going across the WAN. The filter first defines the types of packets that should *not* be filtered:

- AppleTalk Address Resolution Protocol (AARP) packets.
- AppleTalk packets that are not addressed to the AppleTalk multicast address (such as regular traffic related to an actual AppleTalk File Server connection).
- All non-AppleTalk traffic.

The filter then defines the packets that should be dropped:

- AppleTalk Echo Protocol (AEP)
- Name Binding Protocol (NBP)

To define this generic filter:

1 Open a Filter profile and assign it a name. For example:

```
Ethernet
   Filters
      Name=AppleTalk Broadcasts
```

2 Open Output Filters>Out filter 01.

3 Set Valid to Yes and Type to GENERIC.

```
Output filters...
   Out filter 01
      Valid=Yes
       Type=GENERIC
```

Next, configure a filter condition that defines a location, within a packet, that indicates it is an AARP packet. AARP is defined as Ethernet Protocol Type 0x80f3. The filter will prevent the forwarding of outbound AARP packets. Proceed as follows:

1 Set `Offset=14`

This filter condition is applied to every outgoing Ethernet frame. It causes the comparison to start 14 bytes into the Ethernet frame instead of at the beginning.

2 Set `Length=8`

The filter condition compares an eight-byte section of every Ethernet frame.

3 Set `Mask=ffffffffffffffff`

This Mask specifies that every bit of the 8 bytes will be compared.

In some cases, you might want to consider specific bits. Set the Mask parameter with a hexadecimal number that specifies zeroes for the bits that should not be compared.

4 Set `Value=aaaa0300000080f3`

The value 0x80f3 indicates that the packet in the Ethernet frame is an AARP packet.

5 Set `Forward=No`

6 Set `Compare=Equals`

Step 5 and Step 6 ensure that the Pipeline will discard any AARP packet.

7 Set `More=No`

Before continuing, verify the settings of the first sample filter condition:

8 Close this filter.

Now, configure a second filter condition to allow all non-AppleTalk traffic to pass through the Pipeline. AppleTalk is Ethernet Protocol Type 0x809b, so the Pipeline is configured to forward any packet that does not have Type 0x809b. Proceed as follows:

1 Open Out filter 02, and set Valid to Yes and Type to GENERIC.

```
Output filters...
   Out filter 02
      Valid=Yes
      Type=GENERIC
```

2 Set `Offset=14`

This filter condition is applied to every outgoing Ethernet frame. It causes the comparison to start 14 bytes into the Ethernet frame instead of at the beginning.

3 Set `Length=8`

The filter condition compares an eight-byte section of every Ethernet frame.

4 Set `Mask=ffffffffffffffff`

This Mask specifies that every bit of the 8 bytes will be compared.

5 Set `Value=aaaa03080007809b`

The value 0x809b indicates that the packet in the Ethernet frame is an AppleTalk packet.

6 Set `Forward=Yes`

7 Set `Compare=NotEquals`

Step 6 and Step 7 ensure that the Pipeline will forward any non-AppleTalk packet.

8 Set `Forward=Yes`

9 Close this filter.

Next, configure a third filter condition to allow AppleTalk AEP packets to pass through the Pipeline:

1 Open Out filter 03, and set Valid to Yes and Type to GENERIC.

2 Set `Offset=32`

This filter condition is applied to every outgoing Ethernet frame. It causes the comparison to start 32 bytes into the Ethernet frame instead of at the beginning.

3 Set `Length=3`

The filter condition compares a three-byte section of every Ethernet frame.

4 Set `Mask=ffffff0000000000`

This Mask specifies that every bit of the 3 bytes will be compared.

5 Set `Value=0404040000000000`

6 Set `Forward=Yes`

Step 6 and Step 7 ensure that any AEP packet will be forwarded by the Pipeline.

7 Set `Compare=Equals`

8 Close this filter.

Now, configure a filter condition that does not allow broadcast AppleTalk packets to pass through the Pipeline. Take the inverse approach, allowing non-broadcast AppleTalk traffic to pass:

1 Open Out filter 04, and set Valid to Yes and Type to GENERIC.

2 Set `Offset=32`

This filter condition is applied to every outgoing Ethernet frame. It causes the comparison to start 32 bytes into the Ethernet frame instead of at the beginning.

3 Set `Length=6`

The filter condition compares a six-byte section of every Ethernet frame.

4 Set `Mask=ffffffffffff0000`

This Mask indicates that every bit of the 6 bytes will be compared.

5 Set `Value=090007ffffff000`

AppleTalk broadcast traffic is identified by its use of a multicast address. This value specifies a packet that uses a multicast address.

6 Set `Forward=Yes`

Step 6 and Step 7 ensure that the Pipeline will discard any broadcast AppleTalk packet.

7 Set `Compare=NotEquals`

8 Close this filter.

Next, define two linked filter conditions to block Name Binding Protocol (NBP) broadcast packets. These linked output filter conditions specify NBP lookup packets that use a wildcard device name. AppleTalk devices use NBP lookups to search for specific devices. For example, when you open a Macintosh Chooser to connect to a printer, the Macintosh sends NBP lookup packets that are responded to by any available printers. To define the first condition and set up the link:

1 Open Out filter 05, and set Valid to Yes and Type to GENERIC.

2 Set `Offset=32`

This filter condition is applied to every outgoing Ethernet frame. It causes the comparison to start 32 bytes into the Ethernet frame instead of at the beginning.

3 Set `Length=4`

The filter condition compares a four-byte section of every Ethernet frame.

4 Set `Mask=ff00fff000000000`

This Mask indicates that not every bit of the four bytes will be compared.

5 Set `Value=0200022000000000`

6 Set `Forward=Yes`

7 Set `Compare=Equals`

8 Set `More=Yes`

Setting More=Yes in a filter condition links it with the next condition you configure. A packet must match conditions from *both* filter conditions to be forwarded. If a packet satisfies only one condition, the packet is discarded.

9 Close this filter.

To define the second condition:

1. Open Out filter 06, and set Valid to Yes and Type to GENERIC.
2. Set `Offset=42`

   This filter condition is compared to every outgoing Ethernet frame. It causes the comparison to start 42 bytes into the Ethernet frame instead of at the beginning.
3. Set `Length=2`

   The filter condition compares a two-byte section of every Ethernet frame.
4. Set `Mask=ffff000000000000`

   This Mask indicates that every bit of the two bytes will be compared.
5. Set `Value=013d000000000000`
6. Set `Forward=Yes`
7. Set `Compare=Equals`
8. Set `More=No`
9. Close this filter.

Finally, specify that if a packet has passed through the previous filter conditions and has not matched any of them, it will be discarded:

1. Open Out filter 07, and set Valid to Yes and Type to GENERIC.
2. Set `Offset=0`

   This filter condition is applied to every outgoing Ethernet frame. It causes the comparison to start at the beginning of the frame.
3. Set `Length=0`

   The filter condition compares nothing, so every Ethernet frame matches.
4. Set `Mask=0000000000000000`
5. Set `Value=0000000000000000`
6. Set `Forward=No`

   Step 6 and Step 7 ensure that the Pipeline discards any unknown traffic.
7. Set `Compare=Equals`
8. Set `More=No`

An example IP filter to prevent address spoofing

This section shows how to define an IP filter to prevent *spoofing* of local IP addresses. Spoofing IP addresses is a technique whereby outside users pretend to be from the local network in order to obtain unauthorized access to the local network.

The conditions define Input filters that drop packets whose source address either is on the local IP network or is the loopback address (127.0.0.0). All other incoming packets are forwarded.

The filter then defines an Output filter that forwards every outbound packet that has a source address on the local network. All outbound packets with a nonlocal source address are discarded.

Note: This example assumes a local IP network address of 192.100.50.128, with a subnet mask of 255.255.255.192. Use your own local IP address and mask when defining a filter to prevent address spoofing.

To define this IP filter:

1. Open a Filter profile and assign it a name. For example:

```
Ethernet
   Filters
      Name=IP Spoofing
```

2. Open Input Filters>In filter 01.
3. Set `Valid=Yes`
4. Set `Type=IP`

Now configure the first condition to specify the local subnet mask and IP address. If an incoming packet has the specified local address, it will be discarded. Proceed as follows:

1. Set `Protocol=0`
2. Set `Src Port Cmp=None`

   Now the Pipeline will not compare the Port field in the source-address portion of the packet.
3. Set `Src Mask=255.255.255.192`

   The first twenty six bits of the IP address (the first three octets plus two bits from the fourth) indicate the subnet. The remaining six bits indicate the host portion of the address. The Pipeline will compare only the subnet portion of the source address.
4. Set `Src Adrs=192.100.50.128`
5. Set `Dst Port Cmp=None`
6. Set `Dst Mask=0.0.0.0`
7. Set `Dst Adrs=0.0.0.0`
8. Set `Forward=No`
9. Close this filter.

Next, configure a filter condition to discard any incoming packet specifying the loopback address as its source address:

1. Open In filter 02, and set Valid to Yes and Type to IP.

```
Input filters...
   In filter 02
      Valid=Yes
      Type=IP
```

2. Set `Protocol=0`
3. Set `Src Port Cmp=None`

   Now the Pipeline will not compare the Port field in the source-address portion of the packet.
4. Set `Src Mask=255.0.0.0`

   The Pipeline compares the first eight bits of the source address.
5. Set `Src Adrs=127.0.0.0`
6. Set `Dst Port Cmp=None`
7. Set `Dst Mask=0.0.0.0`
8. Set `Dst Adrs=0.0.0.0`
9. Set `Forward=No`

10 Close this filter.

Now configure a filter condition to forward *any* incoming packet (designated as 0.0.0.0) specifying a non-local source address:

1 Open In filter 03, and set Valid to Yes and Type to IP.

```
Input filters...
   In filter 03
      Valid=Yes
       Type=IP
```

2 Set `Protocol=0`

3 Set `Src Port Cmp=None`

4 Set `Src Mask=0.0.0.0`

5 Set `Src Adrs=0.0.0.0`

6 Set `Dst Port Cmp=None`

7 Set `Dst Mask=0.0.0.0`

8 Set `Dst Adrs=0.0.0.0`

9 Set `Forward=Yes`

10 Close this filter and the Input filters subprofile.

Finally, configure an Output Filter to forward any outbound packet with a local source address.

1 Open the Output filters subprofile and select the first Out filter in the list (01).

2 Set Valid to Yes and Type to IP.

```
Output filters...
   Out filter 01
      Valid=Yes
      Type=IP
```

3 Set `Protocol=0`

4 Set `Src Port Cmp=None`

5 Set `Src Mask=255.255.255.192`

6 Set `Src Adrs=192.100.40.128`

7 Set `Dst Port Compare=None`

8 Set `Dst Mask=0.0.0.0`

9 Set `Dst Adrs=0.0.0.0`

10 Set `Forward=Yes`

A sample IP filter for more complex security issues

This section describes an IP filter that illustrates some of the issues you might need to consider when writing your own IP filters. The sample filter does not address intricate points of network security. You might want to use it as a starting point, and augment it to address your security requirements.

In this example, the local network supports a Web server, and the administrator needs to provide WAN access to the server's IP address while restricting WAN access to other hosts on the local network. However, many local IP hosts need to dial out to the Internet and use IP-

based applications such as Telnet or FTP, which means that their response packets must be directed appropriately to the originating host. In this example, the Web server's IP address is 192.9.250.5.

To define this IP filter:

1 Open a Filter profile and assign it a name.

```
Ethernet
   Filters
      Name=Web Safe
```

2 Open Input Filters>In filter 01.

3 Set Valid to Yes and Type to IP.

```
      Input filters...
         In filter 01
            Valid=Yes
            Type=IP
```

Now configure the first condition to forward any packet with a web server's IP address as its destination:

1 Set `Protocol=6`

Protocol 6 specifies TCP packets.

2 Set `TCP Estab=No`

3 Set `Src Port Cmp=None`

4 Set `Src Mask=0.0.0.0`

5 Set `Src Adrs=0.0.0.0`

6 Set `Dst Port Cmp=Eql`

7 Set `Dst Port #=80`

8 Set `Dst Mask=255.255.255.255`

The Pipeline will compare all 32 bits of the destination IP address.

9 Set `Dst Adrs=192.9.250.5`

This is the IP address of the web server.

10 Set `Forward=Yes`

11 Close this filter.

Next, configure a filter condition to forward any incoming TCP packet that uses a source port greater than 1023. The TCP protocol defines these packets as responses to TCP requests. The Pipeline forwards these packets because the initial TCP request was generated by local devices. Local users are allowed to Telnet to remote devices, but remote devices are prevented from establishing TCP or Telnet connections to local devices. Proceed as follows:

1 Open In filter 02, and set Valid to Yes and Type to IP.

```
      Input filters...
         In filter 02
            Valid=Yes
            Type=IP
```

2 Set `Protocol=6`

3 Set `TCP Estab=No`

4 Set `Src Port Cmp=None`

5 Set `Src Mask=0.0.0.0`

6 Set `Src Adrs=0.0.0.0`

7 Set `Dst Port Cmp=Gtr`

8 Set `Dst Port #=1023`

9 Set `Dst Mask=0.0.0.0`

10 Set `Dst Adrs=0.0.0.0`

11 Set `Forward=Yes`

12 Close this filter.

Similarly, configure a filter condition to forward UDP packets. For example, a RIP packet is sent out as a UDP packet to destination port 520. The response to this RIP request is sent to a random destination port greater than 1023. Proceed as follows:

1 Open In filter 03, and set Valid to Yes and Type to IP.

```
Input filters...
   In filter 03
      Valid=Yes
      Type=IP
```

2 Set `Protocol=17`

Protocol 17 specifies UDP packets.

3 Set `Src Port Cmp=None`

4 Set `Src Mask=0.0.0.0`

5 Set `Src Adrs=0.0.0.0`

6 Set `Dst Port Cmp=Gtr`

7 Set `Dst Port #=1023`

8 Set `Dst Mask=0.0.0.0`

9 Set `Dst Adrs=0.0.0.0`

10 Set `Forward=Yes`

11 Close this filter.

Finally, configure a filter condition to forward ICMP packets to allow unrestricted Pings and Traceroutes. ICMP does not use ports like TCP and UDP, so a source and destination port comparison is unnecessary. Proceed as follows:

1 Open In filter 04, and set Valid to Yes and Type to IP.

```
Input filters...
   In filter 04
      Valid=Yes
     Type=IP
```

2 Set `Protocol=1`

Protocol 1 specifies UDP packets.

3 Set `Src Port Cmp=None`

4 Set `Src Mask=0.0.0.0`

5 Set `Src Adrs=0.0.0.0`

6. Set `Dst Port Cmp=None`
7. Set `Dst Mask=0.0.0.0`
8. Set `Dst Adrs=0.0.0.0`
9. Set `Forward=Yes`
10. Exit and save the Filter profile.

Setting Up Virtual Private Networking 12

This chapter covers the following topics:

Introduction to Virtual Private Networking (VPN)

Virtual Private Networks provide low-cost remote access to private LANs via the Internet. The tunnel to the private corporate network can be from an ISP, enabling mobile nodes to dial-in to a corporate network, or it can provide a low-cost Internet connection between two corporate networks. Ascend currently supports two VPN schemes: Ascend Tunnel Management Protocol (ATMP) and Point-to-Point Tunneling Protocol (PPTP).

An ATMP session occurs between two Ascend units via UDP/IP. All packets passing through the tunnel are encapsulated in standard GRE (Generic Routing Encapsulation) as described in RFC 1701. ATMP creates and tears down a cross-Internet tunnel between the two Ascend units. In effect, the tunnel collapses the Internet cloud and provides what looks like direct access to a home network. Bridging is not supported through the tunnels. All packets must be routed with IP or IPX.

Point-to-Point-Tunneling Protocol (PPTP) was developed by Microsoft Corporation to enable Windows 95 and Windows NT Workstation users to dial into a local ISP to connect to a private corporate network across the Internet.

The Pipeline does not support dial-in users, so its support of PPTP consists of routing or forwarding PPTP traffic as appropriate. The Pipeline does not act as either a PPTP Access Concentrator (PAC) or a PPTP Network Server (PNS).

Configuring ATMP tunnels

This section describes how ATMP tunnels work between an Ascend MAX and a Pipeline. The MAX is configured as a *foreign* agent (typically a local ISP) and the Pipeline as a *home* agent, with access to the home network. A mobile node dials into the foreign agent, which establishes a cross-Internet IP connection to the home agent. The foreign agent then requests an ATMP tunnel on top of the IP connection.

The home agent is the terminating part of the tunnel, where most of the ATMP processing occurs. It communicates with the home network (the destination network for mobile nodes) through a direct connection, another router, or across a nailed connection.

For example, in Figure 12-1, the mobile node might be a sales person who logs into an ISP to access his or her home network. The ISP is the foreign agent. The home agent has access to the home network.

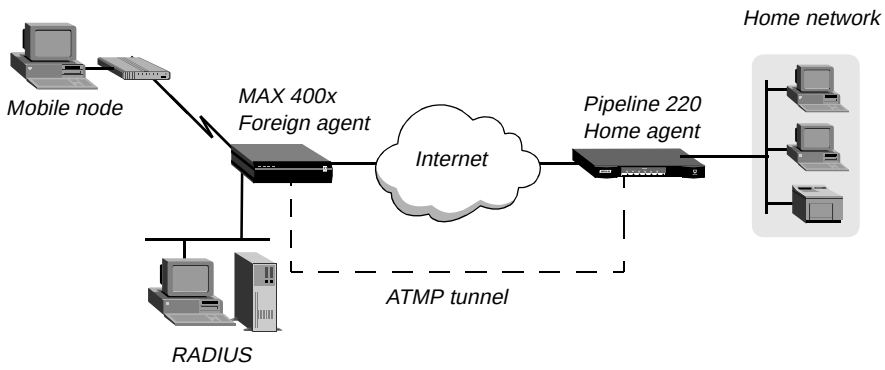


Figure 12-1. ATMP tunnel across the Internet

How the Pipeline creates ATMP tunnels

The mobile node, foreign agent, and home agent establish an ATMP-tunnel connection as follows:

1 A mobile node dials a connection to the foreign agent.

2 After successful authentication, the foreign agent communicates with the home agent, and an IP connection establishes.

3 The foreign agent informs the home agent that the mobile node is connected, and requests a tunnel. It sends up to 10 RegisterRequest messages at 2-second intervals, timing out and logging a message if it receives no response to those requests.

4 The home agent requests a password before it creates the tunnel.

5 The foreign agent returns an encrypted version of the Ascend-Home-Agent-Password found in the mobile node's RADIUS profile. This password must match the home agent's Password parameter in the ATMP configuration in the Ethernet Profile.

6 The home agent returns a RegisterReply with a number that identifies the tunnel. If registration fails, the foreign agent disconnects the mobile node. If registration succeeds, the home agent creates the tunnel between itself and the foreign agent.

7 When the mobile node disconnects from the foreign agent, the foreign agent sends a DeregisterRequest to the home agent to close down the tunnel.

The foreign agent can send its request a maximum of ten times, or until it receives a DeregisterReply. If the foreign agent receives packets for a mobile node whose connection has been terminated, the foreign agent silently discards the packets.

Router and gateway mode

The home agent can communicate with the home network through a direct connection, through another router, or across a nailed connection. When the home agent relies on packet routing to reach the home network, it operates in router mode. When it has a nailed connection to the home network, it is in gateway mode.

Configuring a home agent in router mode

When the ATMP tunnel has been established between the home agent and foreign agent, the home agent in router mode receives IP packets through the tunnel, removes the GRE encapsulation, and passes the packets to its bridge/router software. It also adds to its routing table a host route to the mobile node.

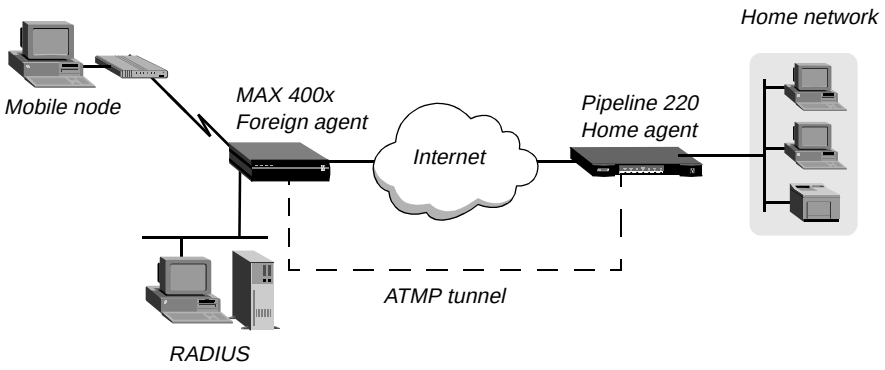


Figure 12-2. Home agent routing to the home network

Understanding the ATMP router mode parameters

This section provides some background information about parameters used in configuring the Pipeline as a home agent in router mode:

Parameter	How it's used
ATMP Mode	To enable the Pipeline's home-agent functionality, ATMP Mode must be set to Home.
ATMP Type	When ATMP Type is set to Router, the home agent relies on routing (not a WAN connection) to pass packets, received through the tunnel, to the home network.
Password	This is the password used to authenticate the ATMP tunnel itself. It must match the password specified in the Ascend-Home-Agent-Password attribute of mobile nodes' RADIUS profiles.
UDP port	By default, ATMP uses UDP port 5150 for ATMP messages between the foreign and home agents. If you specify a different UDP port number, make sure it is consistent between the home agent and foreign agent.
SAP Reply	Enables the home agent to reply to the mobile node's IPX Nearest Server Query if it knows about a server on the home network. If set to No, the home agent simply tunnels the mobile node's request to the home network.
IP configuration and Connection profile	The cross-Internet connection to the foreign agent is an IP routing connection, which is authenticated and established in the usual way. For details, see Chapter 4, "Configuring IP Routing."

Notes about routing to the mobile node

When the home agent receives IP packets through the ATMP tunnel, it adds a host route for the mobile node to its IP routing table. Then it supports IP and IPX routing normally. When the home agent receives IPX packets through the tunnel, it adds a route to the mobile node on the basis of the virtual IPX network number assigned in the RADIUS user profile.

For IP routes, you can enable RIP on the home agent's Ethernet to enable other hosts and networks to route to the mobile node. Enabling RIP is particularly useful if the home network is one or more hops away from the home agent's Ethernet. If RIP is turned off, other routers require static routes that specify the home agent as the route to the mobile node.

Note: If the home agent's Ethernet is the home network (a direct connection), you should turn on proxy ARP in the home agent so that local hosts can use ARP to find the mobile node.

For details about IP routes, see Chapter 4, "Configuring IP Routing." For information about IPX routes, see Chapter 8, "Configuring IPX Routing."

Example of configuring a home agent in router mode (IP)

Before configuring it as a home agent in IP router mode, verify that the Pipeline has a valid IP address in the Ethernet > Mod Config > Ether1 Options (or Ether2 Options) > IP Adrs parameter. You should also validate IP connectivity, by pinging the Pipeline from another IP host.

Configuring system-wide ATMP parameters

To begin configuring the home agent in router mode to reach an IP home network:

1. Open the Ethernet > Mod Config > ATMP Options subprofile.
2. Set ATMP Mode to Home.
3. Set ATMP Type to Router.
4. Specify the password used to authenticate the tunnel (Ascend-Home-Agent-Password).

```
ATMP options...
   ATMP Mode=Home
   Type=Router
   Password=private
   SAP Reply=No
   UDP Port=5150
```

   Note: Set the Password to the value that will be supplied by the RADIUS profile of mobile users. The attribute in the mobile user's profile is Ascend-Home-Agent-Password (unlike Dial-In Password, Ascend-Home-Agent-Password is the same for all users).

5. Exit and save the Ethernet profile.

Configuring a Connection profile to the foreign agent

To configure a Connection profile to provide a route to the foreign agent:

1. Open a Connection profile and configure an IP routing connection to the foreign agent as in the following example:

```
Ethernet
   Connections
```

```
Station=foreign-agent
Active=Yes
Encaps=MPP
Dial #=555-1213
Route IP=Yes

Encaps options...
   Send Auth=CHAP
   Recv PW=foreign-pw
   Send PW=home-pw

IP options...
   LAN Adrs=10.65.212.226/24
```

Note: The Recv PW should be unique for every mobile user, whereas the Password for the ATMP tunnel (in the Ethernet > Mod Config > ATMP Options subprofile) must be identical for all mobile users, and specified in the Ascend-Home-Agent-Password attribute for each user.

2 Exit and save the Connection profile.

Example of configuring a home agent in router mode (IPX)

Before configuring it as a home agent in IPX router mode, verify that the Pipeline must be configured to route IPX. For details, see Chapter 8, “Configuring IPX Routing.”

Configuring system-wide ATMP parameters

To begin configuring the home agent in router mode to reach an IPX network:

1 Open Ethernet>Mod Config>Ether Options and verify that the LAN interface has an IP address (needed to communicate with the foreign agent) and can route IPX.

```
Ethernet
   Mod Config
      IPX Routing=Yes
      Ether options…
         IP Adrs=10.1.2.3/24
         IPX Frame=802.2
         IPX Enet #=00000000
```

For details, see Chapter 8, “Configuring IPX Routing.”

2 Open the ATMP Options subprofile and set ATMP Mode to Home and Type to Router.

3 Specify the password used to authenticate the tunnel (Ascend-Home-Agent-Password).

The attribute in the mobile user’s profile is Ascend-Home-Agent-Password (unlike Dial-In Password, Ascend-Home-Agent-Password is the same for all users).

4 Set SAP Reply to Yes.

```
ATMP options...
   ATMP Mode=Home
   Type=Gateway
   Password=private
   SAP Reply=Yes
   UDP Port=5150
```

5 Exit and save the Ethernet profile.

Configuring a Connection profile to the foreign agent

Now configure a Connection profile to provide a route to the foreign agent:

1 Open a Connection profile and configure an IP routing connection to the foreign agent as in the following example:

```
Ethernet
   Connections
      Station=foreign-agent
      Active=Yes
      Encaps=MPP
      Dial #=555-1213
      Route IP=Yes

      Encaps options...
         Send Auth=CHAP
         Recv PW=foreign-pw
         Send PW=home-pw

      IP options...
         LAN Adrs=10.65.212.226/24
```

Note: The Recv PW should be unique for every mobile user, whereas the Password for the ATMP tunnel (in the Ethernet > Mod Config > ATMP Options subprofile) must be identical for all mobile users, and specified in the Ascend-Home-Agent-Password attribute for each user.

2 Exit and save the Connection profile.

Configuring a home agent in gateway mode

When the home agent is configured in gateway mode, it receives GRE-encapsulated IP packets from the foreign agent, strips off the encapsulation, and passes the packets across a nailed WAN connection to the home network.

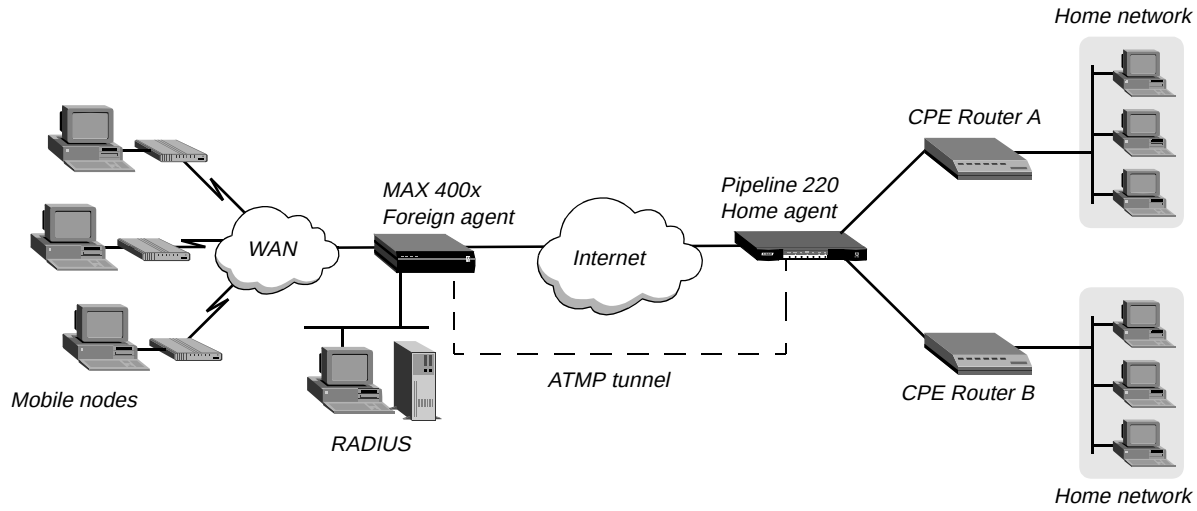


Figure 12-3. Home agent in gateway mode

Note: To enable hosts and routers on the home network to reach the mobile node, you must configure a static route in the Customer Premise Equipment (CPE) router on the home network (not in the home agent). The static route must specify the home agent as the route to the mobile

node. That is, the route's destination address specifies the Framed-Address of the mobile node, and its gateway address specifies the IP address of the home agent.

Understanding the ATMP gateway mode parameters

This section provides some background information about parameters used in configuring the Pipeline as a home agent in gateway mode:

Parameter	How it is used
ATMP Mode	To enable the Pipeline's home agent functionality, ATMP Mode must be set to Home.
Type	When the Type is set to Gateway, the home agent forwards packets received through the tunnel to the home network across a nailed WAN connection.
Password	This is the password used to authenticate the ATMP tunnel itself. It must match the password specified in the Ascend-Home-Agent-Password attribute of mobile nodes' RADIUS profiles.
UDP port	By default, ATMP uses UDP port 5150 for ATMP messages between the foreign and home agents. If you specify a different UDP port number, make sure it is consistent between the home agent and foreign agent.
SAP Reply	Enables the home agent to reply to the mobile node's IPX Nearest Server Query if it knows about a server on the home network. If set to No, the home agent simply tunnels the mobile node's request to the home network.
IP configuration and Connection profile	The cross-Internet connection to the foreign agent is an IP routing connection, which is authenticated and established in the usual way. For details, see Chapter 4, "Configuring IP Routing."
Connection profile to the home network	The Connection profile to the home network must be a local profile. It cannot be specified in RADIUS. The name of this Connection profile must match the name in the Ascend-Home-Network-Name attribute in the mobile node's RADIUS profile.

Example of configuring a home agent in gateway mode (IP)

Before configuring it as a home agent in gateway mode, verify that the Pipeline has a valid IP address in the Ethernet>Mod Config>Ether1 Options (or Ether2 Options) > IP Adrs parameter. You should also validate IP connectivity, by pinging the Pipeline from another IP host.

Configuring system-wide ATMP parameters

To configure the home agent in gateway mode to reach an IP home network:

1 Open Ethernet>Mod Config>Ether1 Options (or Ether2 Options) and verify that the LAN interface has an IP address. For example:

```
Ethernet
   Mod Config
      Ether options...
         IP Adrs=10.1.2.3/24
```

2 Open the ATMP Options subprofile and set ATMP Mode to Home and Type to Gateway.

3 Specify the password used to authenticate the tunnel. This must match the Ascend-Home-Agent-Password attribute of mobile nodes' RADIUS profiles.

```
ATMP options...
   ATMP Mode=Home
   Type=Gateway
   Password=private
   SAP Reply=No
   UDP Port=5150
```

Configuring a Connection profile to the foreign agent

To configure a Connection profile to provide a route to the foreign agent, open a Connection profile and configure an IP routing connection to the foreign agent as in the following example:

```
Ethernet
   Connections
      Station=foreign-agent
      Active=Yes
      Encaps=MPP
      Dial #=555-1213
      Route IP=Yes

      Encaps options...
         Send Auth=CHAP
         Recv PW=foreign-pw
         Send PW=home-pw

      IP options...
         LAN Adrs=10.65.212.226/24
```

Configuring a Connection profile to the home network

To configure a connection profile for the nailed WAN link to the home network, open a Connection profile and configure it as in the following example:

```
Ethernet
   Connections
      Station=homenet
      Active=Yes
      Encaps=MPP
      Dial #=N/A
      Calling #=N/A
      Route IP=Yes

      IP options...
         LAN Adrs=5.9.8.2/24

      Telco options...
         Call Type=Nailed
         Group=1,2

      Session options...
         ATMP Gateway=Yes
```

Example of configuring a home agent in gateway mode (IPX)

Before configuring it as a home agent in IPX router mode, verify that the Pipeline is configured to route IPX. For additional details, see Chapter 8, "Configuring IPX Routing."

Configuring system-wide ATMP parameters

To begin configuring the home agent in gateway mode to reach an IPX network:

1 Open Ethernet>Mod Config>Ether Options and verify that the LAN interface has an IP address (required to communicate with the foreign agent) and can route IPX. For example:

```
Ethernet
   Mod Config
      IPX Routing=Yes
      Ether options…
         IP Adrs=10.1.2.3/24
         IPX Frame=802.2
         IPX Enet #=00000000
```

For details, see Chapter 8, "Configuring IPX Routing."

2 Open the ATMP Options subprofile and set ATMP Mode to Home and Type to Gateway.

3 Specify the password used to authenticate the tunnel. This must match the Ascend-Home-Agent-Password attribute of mobile nodes' RADIUS profiles.

4 Set SAP Reply to Yes.

```
ATMP options...
   ATMP Mode=Home
   Type=Gateway
   Password=private
   SAP Reply=Yes
   UDP Port=5150
```

5 Close the Ethernet profile.

Configuring a Connection profile to the foreign agent

To configure a Connection profile to provide a route to the foreign agent, open a Connection profile and configure it as in the following example:

```
Ethernet
   Connections
      Station=foreign-agent
      Active=Yes
      Encaps=MPP
      Dial #=555-1213
      Route IP=Yes

      Encaps options...
         Send Auth=CHAP
         Recv PW=foreign-pw
         Send PW=home-pw

      IP options...
         LAN Adrs=10.65.212.226/24
```

Configuring a Connection profile to the home network

To configure a connection profile for the nailed WAN link to the home network, open a Connection profile and configure it as in the following example:

```
Ethernet
   Connections
      profile 5...
         Station=homenet
         Active=Yes
         Encaps=MPP
         PRI # Type=National
         Dial #=555-1212
         Route IPX=Yes

         Encaps options...
            Send Auth=CHAP
            Recv PW=homenet-pw
            Send PW=my-pw

         IPX options...
            IPX RIP=None
            IPX SAP=Both
            NetWare t/o=30

         Telco options...
            Call Type=Nailed
            Group=1,2

         Session options...
            ATMP Gateway=Yes
```

SNMP Administrative Support

13

This chapter covers the following topics:

Introduction

The Pipeline supports SNMP on an IP-routed network. An SNMP management station that uses the Ascend Enterprise Management Information Base (MIB) can request information from the Pipeline, set parameters, and send alarm notifications when specific conditions occur in the Pipeline. An SNMP manager must be running on a host on the local IP network, and the Pipeline must be able to find that host, either via a static route or RIP updates.

SNMP supports password security, which you should configure to protect the Pipeline from modification by unauthorized users with access to SNMP management stations.

SNMP traps provide SNMP management stations with real-time system changes. Traps are messages sent notifying SNMP managers of specific events. For example, the Pipeline can notify the SNMP management station that the condition of its nailed link has changed.

Configuring SNMP access security

There are two levels of SNMP security:

- Community strings, which are passwords that you must know for access to the Pipeline.
- Address security, which excludes SNMP access unless it is initiated from a specified IP address.

How the SNMP security options work

The Ethernet > Mod Config > SNMP Options menu provides access to the parameters for configuring community strings and address security.

Enabling read/write access

The R/W Comm Enable parameter enables or disables read/write access to the Pipeline. If you set R/W Comm Enable parameter to No (the default), SNMP managers cannot change the Pipeline configuration via SNMP even if they have the correct read/write community strings.

Community strings

The Read Comm parameter specifies the SNMP community name allowing read access only. The R/W Comm parameter specifies the SNMP community name for read and write access.

Address security

If you set Security to No (the default), any SNMP manager that supplies the configured community name is allowed access to the Pipeline. If you set Security to Yes, the Pipeline allows access only to SNMP managers whose IP addresses are listed in the RD Mgr (for read-only access) or WR Mgr (for read/write access) parameters. You can specify up to five addresses for each type of SNMP manager.

Entering SNMP security settings

You can carry out the following procedure to set the community strings, enforces address security, and prevent write access:

1. Open the Ethernet > Mod Config > SNMP Options submenu.
2. Specify the Read Community Name name as in the following example:

   ```
   Read Comm=public
   ```

3. To enable SNMP read/write access to the Pipeline, set R/W Comm Enable to Yes:

   ```
   R/W Comm Enable=Yes
   ```

4. If required, specify the Read/Write Community name as in the following example:

   ```
   R/W Comm=gzp2Dcj5
   ```

5. If required, enable SNMP security by setting Security to Yes.
6. If required, specify up to five host addresses allowing Read access, and up to five host addresses allowing both Read and Write access.

Setting SNMP traps

A trap is a mechanism for reporting system change in real time (for example, reporting a call coming into to a serial host port). 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 Pipeline with eight separate Trap profiles, directing it to send different combinations of traps to different SNMP managers. For redundancy, you can configure the Pipeline to send identical combinations of traps to different SNMP managers.

Understanding the SNMP trap parameters

To set traps, you use the following parameters:

Parameter	Description
Name	Specifies the name of the Trap profile. If you configure more than one profile, attaching a descriptive name to the trap might help you organize and keep track of different profiles.
Alarm, Port, Security	These parameters specify whether the Pipeline traps alarm events, security events, and port events and sends a trap-PDU to the SNMP manager.
Comm	Used in communicating with the SNMP manager. It must contain the community name associated with the SNMP PDU.
Dest	Specifies the IP address of the system running the SNMP manager. **Note:** To prevent the Pipeline from sending SNMP traps, set Dest=0.0.0.0.

Entering an SNMP trap configuration

You can configure up to eight separate Traps profiles. To configure an SNMP Trap profile:

1. Open an SNMP Traps profile and assign it a name.
2. Specify the types of traps to send to the host, as in the following example:

   ```
   Alarm=Yes
   Port=Yes
   Security=Yes
   ```

3. Specify the community name, as in the following example:

   ```
   Comm=Ascend
   ```

   The Pipeline sends the value you specify to authenticate itself to the SNMP host when an SNMP trap event occurs.

4. Specify the IP address of the host to which the trap-PDUs will be sent, as in the following example:

   ```
   Dest=10.2.3.4
   ```

5 Exit and save the SNMP Traps profile.

Enterprise traps

This section is a brief summary of the traps generated by alarm, port, and security events. For detailed information, see the Ascend Enterprise MIB. For information about obtaining the Ascend MIB, see "Supported MIBs" on page 13-6.

Alarm events

Alarm events (also called *error events*) use trap types defined in RFC 1215 and 1315, in addition to an Ascend enterprise trap type. The Pipeline supports the following trap types from RFC 1215 are supported:

Trap type	Description
coldStart (RFC-1215 trap-type 0)	Signifies that the Pipeline sending the trap is reinitializing itself, so the configuration of the SNMP manager or the unit might be altered.
warmStart (RFC-1215 trap-type 1)	Signifies that the Pipeline sending the trap is reinitializing itself, so that neither the configuration of SNMP manager or the unit is altered.
linkDown (RFC-1215 trap-type 2)	Signifies that the Pipeline sending the trap recognizes a failure in one of the communication links represented in the SNMP manager's configuration.
linkUp (RFC-1215 trap-type 3)	Signifies that the Pipeline sending the trap recognizes that one of the communication links represented in the SNMP manager's configuration has come up.
frDLCIStatusChange (RFC-1315 trap-type 1)	Signifies that the Pipeline sending the trap recognizes that one of the virtual circuits (to which a DLCI number has been assigned) has changed state. That is, the link has either been created, invalidated, or it has toggled between the active and inactive states.
eventTableOverwrite (ascend trap-type 16)	Signifies that a new event has overwritten an unread event. This trap is sent only for systems that 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.

Port state change events

The following traps are effective on a port-by-port basis for each port pointed to by ifIndex. The hostPort objects are used to associate a change with ifIndex objects.

Trap type	Description
portInactive (ascend trap-type 0)	AIM port associated with the passed index has become inactive.
portDualDelay (ascend trap-type 1)	AIM port associated with the passed index is delaying the dialing of a second to avoid overloading devices that cannot handle two calls in close succession.

Trap type	Description
portWaitSerial (ascend trap-type 2)	AIM port associated with the passed index has detected DTR and is waiting for an HDLC controller to come online. CTS is off (V.25 bis dialing only).
portHaveSerial (ascend trap-type 3)	AIM port associated with the passed index is waiting for V.25 bis commands. CTS is on.
portRinging (ascend trap-type 4)	AIM port associated with the passed index has been notified of an incoming call.
portCollectDigits (ascend trap-type 5)	AIM port associated with the passed index is receiving digits from an RS366 interface (RS-366 dialing only).
portWaiting (ascend trap-type 6)	AIM port associated with the passed index is waiting for connect notification from the WAN after dialing or answer notification has been issued.
portConnected (ascend trap-type 7)	AIM port associated with the passed index has changed state. This change of state can be from connected to unconnected or vice versa. If connected to the far end, end-to-end data can flow but has not yet been enabled. The following trap report sequence shows a link is up: portWaiting (6) portConnected (7) portCarrier (8) The following trap report sequence shows a link is down: portConnected (7) portInactive (0)
portCarrier (ascend trap-type 8)	AIM port associated with the passed index has end-to-end data flow enabled.
portLoopback (ascend trap-type 9)	AIM port associated with the passed index has been placed in local loopback mode.
portAcrPending (ascend trap-type 10)	AIM port associated with the passed index has set ACR on the RS366 interface, and is waiting for the host device (RS-366 dialing only).
portDTENotReady (ascend trap-type 11)	AIM port associated with the passed index is waiting for DTE to signal a ready condition when performing X.21 dialing.

Security events

Security events are used to notify users of security problems, and to track access to the unit from the console. The MIB-II event *authenticationError* is a security event. The other security events are Ascend-specific.

Trap type	Description
authenticationFailure (RFC-1215 trap-type 4)	Signifies that the Pipeline sending the trap is the addressee of a protocol message that is not properly authenticated.
consoleStateChange (ascend trap-type 12)	Signifies the console associated with the passed console index has changed state. To read the console's state, get ConsoleEntry from the Ascend enterprise MIB.

Trap type	Description
portUseExceeded (ascend trap-type 13)	The serial host port's use exceeds maximum set by Max DS0 Mins Port parameter associated with the passed index (namely, the interface number).
systemUseExceeded (ascend trap-type 14)	The serial host port's use exceeds maximum set by Max DS0 Mins System parameter associated with the passed index (namely, the interface number).
maxTelnetAttempts (ascend trap-type 15)	There have been three consecutive failed attempts to login onto the Pipeline via Telnet.

Supported MIBs

You can download the most up-to-date version of the Ascend Enterprise MIB by logging in as *anonymous* to ftp.ascend.com. (No password is required.) In addition to the Ascend MIB, the Pipeline also supports objects related to Ascend functionality in the following Internet standard MIBs:

- MIB-II implementation (RFC 1213)
- DS1 MIB implementation (RFC 1406)
- RS232 MIB implementation (RFC 1317)
- Frame Relay MIB implementation (RFC 1315)
- Modem MIB implementation (RFC 1696)

You can download the most recent version of the previous RFCs by logging in as *anonymous* to ftp.ds.internic.net. (No password is required.)

Basic Security Measures

14

This chapter describes how to set up basic security on the Pipeline. If you are using IP Security or Secure Access Firewalls, see the Secure Access Manager (SAM) documentation for information on configuring IP Security and firewalls. This chapter contains these topics:

About Security profiles

A Security profile consists of parameters you can set to control access to the Pipeline. All Security profiles are located below the Security menu of the System profile in the Pipeline configuration interface.

```
00-300 Security
>00-301 Default
 00-302
 00-303
 00-304
 00-305
 00-306
 00-307
 00-308
 00-309 Full Access
```

Two profiles are provided on all Pipeline units:

- Full Access

The Full Access Security profile provides full access to the Pipeline unit. It is the "super-user" profile that enables you to configure your system, dial remote locations, reset the unit, and upgrade system software.

A user who knows the password for the Full Access profile can perform any operation on the Pipeline. The default Full Access password is "Ascend". You should change the Full Access password as soon as possible. For details, see "Changing the Full Access password" on page 14-3.

- Default

  The Pipeline assigns the Default profile to every user who logs in via Telnet, the Control port, and remote management. The Pipeline activates the Default profile whenever the Pipeline powers on or resets. The privileges set in the Default profile are available to all users. You cannot change the name of the Default profile or assign a password to it. However, you can change its settings to make the profile more restrictive. For details, see "Setting the Default profile for read-only access" on page 14-4.

Note: We strongly recommend that you follow the instructions in "Changing the Full Access password" on page 14-3 and "Setting the Default profile for read-only access" on page 14-4. These instructions result in two security levels, one that is totally open (Full Access) and one that is totally restrictive (Default).

If you are the only user who must configure the Pipeline or perform administrative tasks, you do not need to create any Security profiles in addition to the ones provided. However, many sites choose to define additional security levels and enable certain users to perform a subset of administrative functions. You can create up to seven additional Security profiles. For more information on these tasks, see Chapter 2, "Setting Up Security Profiles."

Understanding basic security measures

When the Pipeline is shipped from the factory, all levels are set with full privileges. A profile must have a name to be activated, so only the Default and Full Access profiles can be activated initially. Their default security settings enable you to configure and set up the Pipeline without any restrictions. Before you make the Pipeline generally accessible, you should change these default security settings to protect the configured unit from unauthorized access. These are the steps you must carry out:

1. Change the Full Access password.
2. Activate the Full Access profile
3. Set the Default profile for read-only access.
4. Change the SNMP read-write community string.
5. Assign a Telnet password.
6. Require profiles for incoming connections.
7. Turn off ICMP redirects.

Each section that follows describes each of these steps.

Changing the Full Access password

The Full Access Security profile is the “super-user” profile that enables you to configure your system, dial remote locations, reset the unit, and upgrade system software. This profile is intended to remain totally open, with all privileges set to Yes. The default password assigned to the profile is “Ascend.” A user who knows the password for the Full Access profile can perform any operation on the Pipeline.

Change the default password as soon as possible. And remember: *do not* to turn off the Edit Security privilege in the Full Access profile, or you will be unable to edit privileges when you activate Full Access.

To assign a password protecting the Full Access profile, follow these steps:

1. Open the System > Security menu.
2. Open the Full Access profile.
3. When prompted, enter the default password:

   Ascend

   Passwords are case sensitive. You must enter the password exactly as shown.
4. Select the Passwd parameter and press Enter to open a text field.
5. Type a new password for the profile.
6. Press Enter.

   The string “*SECURE*” replaces the letters you typed, as in:

```
00-309 Full Access
 Name=Full Access
>Passwd=*SECURE*
 Operations=Yes
 Edit Security=Yes
 Edit System=Yes
 Edit Line=Yes
 Edit All Ports=Yes
 Edit Own Port=N/A
 Edit All Calls=Yes
 Edit Com Call=N/A
 Edit Own Call=N/A
 Edit Cur Call=N/A
 Sys Diag=Yes
 All Port Diag=Yes
 Own Port Diag=N/A
 Download=Yes
 Upload=Yes
 Field Service=Yes
```

7. Leave all other privileges enabled.

   Do not turn off the Edit Security privilege!
8. Exit the Full Access profile, saving your changes.

Activating the Full Access profile

You must activate the Full Access profile for your own use in performing the rest of the basic security measures. To activate the Full Access profile, follow these steps:

1 Press Ctrl-D to open the DO menu, and then press P (or select P=Password).

```
00-300 Security
DO…
>0=ESC
 P=Password
```

2 In the list of Security profiles that opens, select Full Access.

The Pipeline prompts you for the Full Access password:

```
00-300 Security
Enter Password:
 []

 Press > to accept
```

3 Type the password assigned to the profile and press Enter.

When you enter the correct password, the Pipeline displays a message informing you that the password was accepted and that the Pipeline is using the new security level

```
Message #119
Password accepted.
Using new security level.
```

If the password you enter is incorrect, the Pipeline prompts you again for the password.

Setting the Default profile for read-only access

The first profile in the Security menu is named Default. The password assigned to this profile is null, and the profile's name and password cannot be changed. The Pipeline activates this profile whenever you power on or reset the unit, and whenever a user begins a new login session.

Although the Default profile is set initially with full privileges, it is intended to be very restrictive. Every user who logs in via Telnet, the Control port, or remote management is granted the privileges specified there.

To make the Default profile appropriately restrictive, follow these steps:

1 Open the System > Security menu.

2 Open the Default profile.

The first two parameters in the Default profile cannot be changed—the name is always Default and the password is always null.

3 Set Operations=No.

```
00-301 Default
 Name=Default
 Passwd=
>Operations=No
 Edit Security=N/A
 Edit System=N/A
```

```
Edit Line=N/A
Edit All Ports=N/A
Edit Own Port=N/A
Edit All Calls=N/A
Edit Com Call=N/A
Edit Own Call=N/A
Edit Cur Call=N/A
Sys Diag=N/A
All Port Diag=N/A
Own Port Diag=N/A
Download=N/A
Upload=N/A
Field Service=N/A
```

All other parameters are set to N/A when Operations=No.

From now on, users who access the Pipeline terminal server cannot make any changes to its configuration or to perform restricted operations. For all users with the Default security level, passwords (including the null password) are hidden by the string *SECURE* in the Pipeline unit's user interface.

4 Exit the Default profile, saving your changes.

Configuring SNMP security

An SNMP community string is an identifier that an SNMP manager application must specify before it can access the MIB (Management Information Base). The Pipeline has two community strings:

- Read Comm

  The read community string has the value "public" by default. It enables an SNMP manager to perform read commands (get and get next) in order to request specific information.

- R/W Comm

  The read-write community string has the value "write" by default. It enables an SNMP manager to perform both read and write commands (get, get next, and set). Using these commands, the application can access management information, set alarm thresholds, and change settings on the Pipeline.

By default SNMP write access to the Pipeline is disallowed. To enable it, set the R/W Comm Enable parameter to Yes.

To configure SNMP security, proceed as in the following example:

1 Open the Ethernet > Mod Config > SNMP Options menu.

2 To enable SNMP write access, set R/W Comm Enable to Yes:

```
R/W Comm Enable=Yes
```

3 For the R/W Comm parameter, specify a text string containing up to 16 characters.

For example, you can specify this setting:

```
R/W Comm=unique-string
```

4 Close the SNMP Options menu, saving your changes.

Assigning a Telnet password

Until you assign a Telnet password, any local user who knows the Pipeline unit's IP address can start a Telnet session with the Pipeline. When you assign a password, all users requesting incoming Telnet sessions, whether locally or from across the WAN, must enter the password.

To assign a Telnet password, follow these steps:

1. Open the Ethernet > Mod Config > Ether Options menu.
2. For the Telnet PW parameter, specify a password containing up to 20 characters.
   For example, you might enter this setting:
   `Telnet PW=telnet-pwd`
3. Close the Ether Options menu, saving your changes.

Requiring profiles for incoming connections

You can use the Pipeline unit's Answer profile to build connections that do not require a name and password. Although some sites allow such connections, most sites impose much tighter restrictions. You should strongly consider limiting incoming connections to those that have a configured Connection profile.

To require configured profiles for all incoming connections, follow these steps:

1. Open the Ethernet > Answer menu.
2. To specify that a matching profile is required for incoming calls, set Profile Reqd to Yes.

   Note: If you support ARA (AppleTalk Remote Access) connections through the Pipeline, setting Profile Reqd to Yes disables Guest access to your network.
3. Save your changes.

Turning off ICMP redirects

ICMP enables a unit to find the most efficient IP route to a destination. ICMP Redirect packets are one of the oldest route discovery methods on the Internet and one of the least secure; it is possible to counterfeit ICMP Redirects and change the way a device routes packets. If the Pipeline is routing IP, we recommend that you turn off ICMP redirects.

To configure the Pipeline to ignore ICMP redirect packets, follow these steps:

1. Open the Ethernet > Mod Config menu.
2. Set ICMP Redirects to Ignore.
3. Save your changes.

Configuring a Security profile

To configure a Security profile, follow these steps:

1 Open the System > Security menu.

2 Open an unnamed profile.

3 For the Name parameter, specify a name for the profile.

You can enter up to 16 characters. For example, you might specify this setting:

Name=Calabasas

4 For the Passwd parameter, specify a password containing up to 20 characters.

As soon as you press Enter, the Pipeline hides the password string you specified by displaying the string "*SECURE*".

5 To enable or disable read-only security, set the Operations parameter.

Yes enables a user to view Pipeline profiles and to change the value of any parameter. The default value is Yes.

No permits a user to view Pipeline profiles, but not to change the value of any parameter. If you specify No, a user cannot access most DO commands. Only DO Esc, DO Close Telnet, and DO password are available.

6 To grant or restrict privileges to edit Security profiles, set the Edit Security parameter.

Yes grants privileges. When you specify Yes, a user can edit Security profiles, and can access all other operations by enabling them in his or her active Security profile. In addition, all passwords in Security profiles are visible as text. This privilege is the most powerful one you can assign, because it allows users to change their own privileges at will. The default value is Yes.

No restricts privileges. When Edit Security=No, all passwords are hidden by the string "*SECURE*."

Note: Do not set the Edit Security parameter to No on all nine Security profiles; if you do, you cannot edit any of them.

7 To grant or restrict privileges to edt the System profile and the Ethernet profile, set the Edit System parameter.

Yes grants privileges to edit the System profile, and to edit the Read Comm and R/W Comm parameters in the Ethernet profile. The default value is Yes.

No restricts edit privileges.

8 To indicate whether an operator can perform all system diagnostics, set the Sys Diag parameter.

Yes specifies that an operator can use any of the options in the Sys Diag menu by local or remote management. The default value is Yes.

No specifies that an operator cannot use any of the options in the Sys Diag menu.

9 To indicate whether an operator can download the configuration of the Pipeline using the Save Cfg command, set the Download parameter.

Yes specifies that a user can download profiles and other configuration parameters to another device for backup. The default value is Yes.

No specifies that an operator cannot download profiles and other configuration parameters.

Note: Whether you choose Yes or No, you cannot download passwords to another device.

10 To indicate whether an operator can upload the Pipeline configuration from another device using the Restore Cfg command, set the Upload parameter.

Yes specifies that the user can upload profiles and other configuration parameters from another device to the Pipeline. You must set Upload=Yes in order to use the Restore Cfg command. The default value is Yes.

No specifies that the user cannot upload profiles and other configuration parameters from another device to the Pipeline.

Note: When you save a configuration to file, passwords are not included in the download, so restoring from file clears all passwords on the Pipeline.

11 To grant or restrict privileges to perform Ascend-provided field service operations, such as uploading new system software, set the Field Service parameter.

Yes grants privileges. The default value is Yes.

No restricts privileges. Selecting No does not disable access to any Pipeline operations. Field service operations are special diagnostic routines not available through Pipeline menus.

12 Close the new Security profile.

Note that the Pipeline does not support all the parameters in the Security profiles.

Activating a Security profile

When you log into the Pipeline, you can only view settings, because the Default profile is active. To make any changes or perform any administrative tasks, you must activate the Full Access profile or any other profile configured to allow setup or administrative tasks.

To activate a profile, follow these steps:

1 Press Ctrl-D to open the DO menu

2 Press P, or select P=Password.

3 In the list of Security profiles that opens, select the profile you want to activate.

The Pipeline prompts you for the password.

4 Specify the appropriate password, and press Enter.

When you enter the correct password, the Pipeline displays a message informing you that the password was accepted and that the Pipeline is using the new security level. If the password you enter is incorrect, the Pipeline prompts you again for the password.

Upgrading System Software

A

Caution: You must use the new software loading procedure explained in ”Upgrading system software” to load this version of software onto your system. Read the instructions carefully before upgrading your system.

If you are upgrading your software using TFTP, you must use the `fsave` command immediately after executing the `tload` command. Failure to do so may cause your Ascend unit to lose its configuration.

Each incremental release contains new features and corrections. To use this release note:

1. Read through the table of contents to determine which software release and (new features) apply to your environment.
2. Obtain the file from Ascend anonymous FTP server (ftp.ascend.com). If you need Technical Assistance, 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)
- UK	(+33) 492 96 5671
- Germany/Austria/Switzerland	(+33) 492 96 5672
- France	(+33) 492 96 5673
- Benelux	(+33) 492 96 5674
- Spain/Portugal	(+33) 492 96 5675
- Italy	(+33) 492 96 5676
- Scandinavia	(+33) 492 96 5677
- Middle East and Africa	(+33) 492 96 5679
E-mail	support@ascend.com
E-mail (outside US)	EMEAsupport@ascend.com
Facsimile (FAX)	510-814-2312
Customer Support BBS by modem	510-814-2302

3. Upgrade to the new software by following the instructions in the next section, “Upgrading system software.” Then configure the features that apply to your site.

Upgrading system software

Caution: The procedure for uploading new software to Ascend units have changed significantly. Carefully read the new software loading procedures explained in this section before upgrading your system.

This section explains how to upgrade your system software. It contains the following sections:

- Definitions and terms
- Guidelines for upgrading system software
- Before you begin
- Upgrading system software with a standard load
- Using the serial port to upgrade to a standard or a thin load
- System messages

Definitions and terms

This document uses the following terms:

Build	The name of the software binary. For example, `ti.m40` is the MAX 4000 T1 IP-only software build. For the names of all the software builds and the features they provide see `/pub/Software-Releases/Max/SW-Filenames-Max.txt` or `/pub/Software-Releases/Pipeline/SW-Filenames-Pipeline.txt` on the Ascend FTP server. If possible, you should stay with the same build when upgrading. Loading a different build can cause your Ascend unit to lose its configuration. If this happens, you must restore your configuration from a backup.
Standard load	Software versions 4.6Ci18 or earlier and all 4.6Cp releases. You can load these versions of software through the serial port or by using TFTP. TFTP is the recommended upgrade method for standard loads.
Fat load	4.6Ci19 to 5.0Aix and all 5.0Ap releases with a file size greater than 960 KB (for MAX units) or 450K (for Pipeline units). Before upgrading to a fat load for the first time, you must upgrade to a thin load. You must use TFTP to upgrade to fat loads. **Note:** The Pipeline 220 does not have fat loads. All its loads are thin.
Thin load	4.6Ci19 to 5.0Aix and all 5.0Ap releases with a file size less than 960 KB (for MAX units) or 450 KB (for Pipeline units). TFTP is the recommended upgrade method for thin loads. **Note:** The Pipeline 220 does not have fat loads. All its loads are thin.

Guidelines for upgrading system software

Caution: Before upgrading, consider the following very important guidelines:

- Use TFTP to upgrade if possible. TFTP is more reliable and saves the Ascend unit configuration when you upgrade.
- You cannot load a fat load through the serial port. You must use TFTP.
- If you are using TFTP to upgrade your software, use the `fsave` command immediately after executing the `tload` command. Failure to do so might cause your Ascend unit to lose its configuration.
- If possible, you should always stay with the same build of software when you upgrade. If you load a different version, your Ascend unit may lose its configuration. If this happens, you must restore your configuration from a backup.
- If you are upgrading to a software version 5.0A or 5.0Aix fat load for the first time, you must be on a load that supports the fat load format. All versions of software 5.0A or above support fat loads. You should perform the upgrade in two steps:
  - Upgrade to a thin load of the same build
  - Upgrade to the fat load
- You can upgrade to a thin load from any version of software.

Before you begin

Make sure you perform all the tasks explained in Table A-1 before upgrading your software.

Table A-1. Before upgrading

Task	Description
If necessary, activate a Security Profile that allows for field upgrade.	If you are not sure how, see the section about Security Profiles in your documentation.
Record all of the passwords you want to retain, and save your Ascend unit's current configuration to your computer's hard disk.	For security reasons, passwords are not written to configuration files created through the serial console. A configuration file created using the Tsave command, however, *does* contain the system passwords. You can restore the Tsave configuration file using the serial console. If you chose to save your configuration using the serial console, you will have to restore your passwords manually. Restoring passwords is explained in "Using the serial port to upgrade to a standard or a thin load" on page A-5.

Table A-1. Before upgrading (continued)

Task	Description
Obtain the correct file, either by downloading it from the FTP server or by requesting it from Ascend technical support.	To ensure that you load the correct software binary, you should check the load currently installed on your unit. To do so: **1** Tab over to the System status window. **2** Press Enter to open the Sys Options menu. **3** Using the Down-Arrow key (or Ctrl- N), scroll down until you see a line similar to the following: `Load: tb.m40` **4** When upgrading, obtain the file with same name from the Ascend FTP site. If your unit does not display the current load or you are unsure about which load to use, contact technical support.
If you are using TFTP, make sure you load the correct binaries into the TFTP home directory on the TFTP server.	You must use TFTP to upgrade to a fat load load.
If you are using the serial port, make sure you have a reliable terminal emulation program, such as Procomm Plus.	If you use the serial port, you can only upgrade to a standard or a thin load. Upgrading through the serial port is not recommended. If you use a Windows-based terminal emulator such as Windows Terminal or HyperTerminal, disable any screen savers or other programs or applications that could interrupt the file transfer. Failure to do so might cause the software upload to halt, and can render the Ascend unit unusable.

Upgrading system software with a standard load

To upgrade system software with a standard load you can use either the serial port or TFTP. TFTP is the recommended method because it preserves your Ascend unit's configuration. If you want to use the serial port to upgrade, see "Using the serial port to upgrade to a standard or a thin load" on page A-5.

Using TFTP to upgrade to a standard load

To upgrade to a standard load using TFTP, you only have to enter a few commands. But you must enter them in the correct sequence, or you could lose the Ascend unit's configuration.

To upgrade to a standard load via TFTP:

1 Obtain the software version you want to upgrade to and place it in the TFTP server home directory.

2 From the Ascend unit's VT100 interface, access the diagnostics monitor by typing the following characters in rapid succession:

Esc [Esc =

Or, press Ctrl-D to invoke the DO menu and select D=Diagnostics.

3 At the > prompt, use the Tsave command to save your configuration as in the following example:

> tsave tftp-server router1.cfg

This saves the configuration of your unit to the file named `router1.cfg` in the TFTP home directory of the server named `tftp-server`. This file must already exist and be writable. Normally, TFTP upgrades save the configuration. Tsave is a precaution.

Caution: The file you save with the Tsave command contains all the passwords in clear text. You should move this file from the TFTP directory to a secure location after the upgrade procedure is complete.

4 Enter the following command:

tloadcode *hostname filename*

where *hostname* is the name or IP address of your TFTP server, and *filename* is the name of the system software on the server (relative to the TFTP home directory).

For example, the command:

```
tloadcode tftp-server t.m40
```

loads `t.m40` into flash from the machine named `tftp-server`.

Caution: You must use the Fsave command immediately after executing the Tload command. Failure to do so can cause your Ascend unit to lose its configuration.

5 Enter the following command to save your configuration to flash memory:

fsave

6 Enter the following command:

nvramclear

After the Ascend unit clears NVRAM memory, it automatically resets.

This completes the upgrade.

Using the serial port to upgrade to a standard or a thin load

Caution: Uploading system software via the serial console overwrites all existing profiles. Save your current profiles settings to your hard disk before you begin upgrading system software. After the upgrade, restore your profiles from the backup file you created. Since the backup file is readable text, you can reenter the settings through the Ascend unit's user interface. To avoid having existing profiles overwritten, use TFTP to upgrade your unit.

Caution: You cannot upload a fat load using the serial port; it must be done using TFTP.

Upgrading through the serial port consists of the following general steps:

- Saving your configuration
- Uploading the software
- Restoring the configuration

Before you begin

Before upgrading your system through the serial port, make sure you have the following equipment and software:

- An IBM compatible PC or Macintosh with a serial port capable of connecting to the Ascend unit's Console port.
- A straight-through serial cable.
- Data communications software for your PC or Mac with XModem CRC/1K support (for example, Procomm Plus, HyperTerminal for PCs or ZTerm for the Mac).

Caution: If you use a Windows-based terminal emulator such as Windows Terminal or HyperTerminal, disable any screen savers or other programs or applications that could interrupt the file transfer. Failure to do so might cause the software upload to halt, and can render the Ascend unit unusable.

Saving your configuration

Before you start, verify that your terminal emulation program has a disk capture feature. Disk capture allows your emulator to capture to disk the ASCII characters it receives at its serial port. You should also verify that the data rate of your terminal emulation program is set to the same rate as the Term Rate parameter in the System Profile (Sys Config menu).

You can cancel the backup process at any time by pressing Ctrl-C.

To save the Pipeline configuration (except passwords) to disk:

1 Open the Sys Diag menu.

2 Select Save Config, and press Enter.

The following message appears:

Ready to download - type any key to start....

3 Turn on the Capture feature of your communications program, and supply a filename for the saved profiles. (Consult the documentation for your communications program if you have any questions about how to turn on the Capture feature.)

4 Press any key to start saving your configured profiles.

Rows of configuration information appear on the screen as the configuration file is downloaded to your hard disk. When the file has been saved, your communications program displays a message indicating the download is complete.

5 Turn off the Capture feature of your communications program.

6 Print a copy of your configured profiles for later reference.

You should examine the saved configuration file. Notice that some of the lines begin with `START=` and other lines begin with `END=`. A pair of these START/STOP lines and the block of data between them constitute a profile. If a parameter in a profile is set to its default value, it does not appear. In fact, you can have profiles with all parameters at their defaults, in which case the corresponding START/STOP blocks are empty. Make sure that there are no extra lines of text or characters either before `START=` or after `END=`. If there are, delete them. They could cause problems when you try to upload the file to the Ascend unit.

Uploading the software

To upload the software:

1. Type the following four-key sequence in rapid succession (press each key in the sequence shown, one after the other, as quickly as possible):

   Esc [Esc -

   (Press the escape key, the left bracket key, the escape key, and the minus key, in that order, in rapid succession.) The following string of Xmodem control characters appears:

   CKCKCKCK

   If you do not see these characters, you probably did not press the four-key sequence quickly enough. Try again. Most people use both hands and keep one finger on the escape key.

2. Use the Xmodem file-transfer protocol to send the system file to the Ascend unit.

   Your communications program normally takes anywhere from 5 to 15 minutes to send the file to your Ascend unit. The time displayed on the screen does not represent real time. Do not worry if your communication program displays several “bad batch” messages. This is normal.

After the upload, the Ascend unit resets. Upon completion of the self-test, the Ascend unit’s initial menu appears in the Edit window with all parameters set to default values. This completes the upgrade.

If the upload fails during the transfer, try downloading another copy of the binary image from the Ascend FTP server and re-loading the code to the Ascend unit. If you still have problems, contact Ascend technical support for assistance.

Restoring the configuration

Under certain circumstances, the serial-port method might not completely restore your configuration. You should therefore verify that your configuration was properly restored every time you use this method. If you have many profiles and passwords, you should consider using TFTP to upgrade your software. (See “Using TFTP to upgrade to a standard load” on page A-4.)

To restore the configuration, you must have administrative privileges that include Field Service (such as the Full Access Profile, for example). You use the Restore Cfg command to restore a full configuration that you saved by using the Save Cfg command, or to upload more specific configuration information obtained from Ascend (for example, a single filter stored in a special configuration file).

To load configuration information through the serial port

1. From the Ascend unit’s VT100 interface, access the diagnostics monitor by typing the following characters in rapid succession:

   Esc [Esc =

   Or, press Ctrl-D to invoke the DO menu, and select D=Diagnostics.

2. At the **>** prompt, enter the Fclear command:

   > **`fclear`**

3. At the **>** prompt, enter the NVRAMClear command:

   > **`nvramclear`**

This causes the system to reset. When it comes back up, proceed with restoring your configuration.

4 Enter **`quit`** to exit the Diagnostic interface.

5 Open the Sys Diag menu.

6 Select Restore Cfg, and press Enter.

The following message appears:

```
Waiting for upload data...
```

7 Use the Send ASCII File feature of the communications software to send the configuration file to the unit. (If you have any questions about how to send an ASCII file, consult the documentation for your communications program.)

When the restore has been completed, the following message appears:

```
Restore complete - type any key to return to menu
```

8 Press any key to return to the configuration menus.

9 Reset the Ascend unit, by selecting System > Sys Diag > Sys Reset and confirming the reset.

Restoring passwords

For security reasons, passwords are not written to configuration files created through the serial console. A configuration file created using the Tsave command, however, *does* contain the system passwords. You can restore the Tsave configuration file using the serial console.

After upgrading you may have to re-enter all the passwords on your system. If you edit your saved configuration file, however, and enter passwords in the appropriate fields (by replacing the word *SECURE* in each instance), these passwords will be restored. But note that if you do choose to edit your configuration file, you must save it as text only or you will not be able to load it into your unit.

If you restored a complete configuration, the passwords used in your Security profiles have been wiped out. To reset them:

1 Press Ctrl-D to invoke the DO menu, select Password, and choose the Full Access profile.

2 When you are prompted to enter the password, press Enter (the null password).

After you have restored your privileges by entering the null password, you should immediately open the Connection profiles, Security profiles, and Ethernet profile (Mod Config menu), and reset the passwords to their previous values.

System messages

Table A-2 explains the messages that can appear during your upgrade.

Table A-2. System software messages

Message	Explanation
`This load has no platform identifier.  Proceed with caution.`	This message can occur if you are running software version 5.0Ai11 or later and you load an earlier incremental or patch release onto your system. The message indicates that Tloadcode cannot determine which platform the code is intended for. If you are using the correct software version, you can ignore this message.
`This load appears not to support your network interface.` `Download aborted.  Use 'tloadcode -f' to force.`	Indicates you are attempting to load a version of code intended for a different network interface (for example, loading MAX 4000 T1 software onto a MAX 4000 E1 unit).
`This load appears to be for another platform.` `Download aborted.  Use 'tloadcode -f' to force.`	Indicates you are attempting to load a version of code onto a platform for which it is not intended (for example, loading MAX 4000 software onto a MAX 2000). This is not recommended

`UART initialized`
`fat load: inflate`
`.....`
`.....`
`starting system...`
	Indicates you have successfully loaded a fat load.

`UART initialized`
`extended load:`
`inflate essential`
`.+.+.....`
`inflate expendable`
`.....|.....`
`starting system...`
	Indicates you have successfully loaded an extended load.

`UART initialized`
`thin load: inflate`
`.....`
`.....`
`starting system...`
	Indicates you have successfully loaded a thin load.

Warranties and FCC regulations

B

Product warranty

1 Ascend Communications, Inc. warrants that the Pipeline 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.

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.

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
04DU9-BN	P220-T1	6.0N	RJ48C
04DU9-DN	P220-T1	6.0N	RJ48C
04DU9-1KN	P220-T1	6.0N	RJ48C
04DU9-1SN	P220-T1	6.0N	RJ48C
04DU9-1ZN	P220-T1	6.0N	RJ48C

IC CS-03 Notice

The Industry Canada label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective, operational and safety requirements as prescribed in the appropriate Terminal Equipment Technical Requirements document(s). 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. The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs to certified equipment should be coordinated by a representative 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 to rural areas.

Caution: Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

INDEX

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

Z