Configuring the PA-VXA, PA-VXB, and PA-VXC

To continue your PA-VXA, PA-VXB, or PA-VXC port adapter installation, you must configure the card type as either T1 or E1 and then configure the interfaces. The instructions that follow apply to all supported platforms. Minor differences among the platforms—with Cisco IOS software commands—are noted.

This chapter contains the following sections:

Using the EXEC Command Interpreter

You modify the configuration of your router through the software command interpreter called the *EXEC* (also called enable mode). You must enter the privileged level of the EXEC command interpreter with the **enable** command before you can use the **configure** command to configure a new interface or change the existing configuration of an interface. The system prompts you for a password if one has been set.

The system prompt for the privileged level ends with a pound sign (#) instead of an angle bracket (>). At the console terminal, use the following procedure to enter the privileged level:

Step 1 At the user-level EXEC prompt, enter the **enable** command. The EXEC prompts you for a privileged-level password as follows:

```
Router> enable

Password:
```

Step 2 Enter the password (the password is case sensitive). For security purposes, the password is not displayed.

When you enter the correct password, the system displays the privileged-level system prompt (#):

```
Router#
```

For complete descriptions of software configuration commands, refer to the publications listed in the "Related Documentation" section on page viii.

Configuring the Interface

After you verify that the new PA-VXA, PA-VXB, or PA-VXC is installed correctly (the enabled LED goes on), use the privileged-level **configure** command to configure the new interface. Have the following information available:

- Protocols you plan to route on the interface
- IP addresses, if you plan to configure the interface for IP routing
- Bridging protocols you plan to use

If you installed a new PA-VXA, PA-VXB, or PA-VXC or if you want to change the configuration of an existing interface, you must enter configuration mode to configure the new interface. If you replaced a PA-VXA, PA-VXB, or PA-VXC that was previously configured, the system recognizes the new interface and brings it up in its existing configuration.

For a summary of the configuration options available and instructions for configuring the interface on a PA-VXA, PA-VXB, or PA-VXC, refer to the appropriate configuration publications listed in the "Related Documentation" section on page viii.

You execute configuration commands from the privileged level of the EXEC command interpreter, which usually requires password access. Contact your system administrator, if necessary, to obtain password access. (See the "Using the EXEC Command Interpreter" section on page 4-1 for an explanation of the privileged level of the EXEC.)

This section contains the following subsections:

Shutting Down a Controller

Before you remove a PA-VXA, PA-VXB, or PA-VXC that you will not replace, or replace a PA-VXA, PA-VXB, or PA-VXC, use the **shutdown** command to shut down (disable) the controller to prevent anomalies when you reinstall the new or reconfigured port adapter. When you shut down the controller, it is designated *administratively down* in the **show** command displays.

Follow these steps to shut down an interface:

Step 1 Enter the privileged level of the EXEC command interpreter (also called enable mode). (See the "Using the EXEC Command Interpreter" section on page 4-1 for instructions.)

Step 2 At the privileged-level prompt, enter configuration mode and specify that the console terminal is the source of the configuration subcommands, as follows:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

Step 3 Shut down the controller by entering the **controller t1** or **controller e1** subcommand (followed by the interface address of the controller), and then enter the **shutdown** command. Table 4-1 shows the command syntax.

When you have finished, press **Ctrl-Z**—hold down the **Control** key while you press **Z**—or enter **end** or **exit** to exit configuration mode and return to the EXEC command interpreter.

Table 4-1 Syntax of the shutdown Command

Platform	Command	Example

Cisco 7200 series and Cisco 7200 VXR routers	**controller**, followed by the *type* (**t1** or **e1**) and *slot/port* (port-adapter-slot-number/ interface-port-number)	The example is for the T1 controller in port adapter slot 1.
		`Router(config)# controller t1 1/0`
		`Router(config)# shutdown`
		`Ctrl-Z`
		`Router#`

Cisco 7301 routers	**controller**, followed by the *type* (**t1** or **e1**) and *slot/port* (port-adapter-slot-number/ interface-port-number)	The example is for the T1 controller in port adapter slot 1.
		`Router(config)# controller t1 1/0`
		`Router(config)# shutdown`
		`Ctrl-Z`
		`Router#`

Cisco 7401ASR routers	**controller**, followed by the *type* (**t1** or **e1**) and *slot/port* (port-adapter-slot-number/ interface-port-number)	The example is for the T1 controller in port adapter slot 1.
		`Router(config)# controller t1 1/0`
		`Router(config)# shutdown`
		`Ctrl-Z`
		`Router#`

VIP in Cisco 7500 series routers	**controller**, followed by the *type* (**t1** or **e1**) and *slot/port adapter/port* (interface-processor-slot-number/ port-adapter-slot-number/ interface-port-number)	The example is for the T1 controller in port adapter slot 0 of a VIP installed in interface processor slot 1.
		`Router(config-if)# controller t1 1/0/0`
		`Router(config-if)# shutdown`
		`Ctrl-Z`
		`Router#`

Step 4 Write the new configuration to NVRAM as follows:

```
Router# copy running-config startup-config
[OK]
Router#
```

The system displays an OK message when the configuration has been stored in NVRAM.

Step 5 Verify that the new interface is now in the correct state (shut down) using the **show controllers** command (followed by the interface type and interface address of the interface) to display the specific interface.

Step 6 Reenable the interface by doing the following:

a. Repeat Step 3 to reenable an interface. Substitute the **no shutdown** command for the **shutdown** command.

b. Repeat Step 4 to write the new configuration to memory. Use the **copy running-config startup-config** command.

c. Repeat Step 5 to verify that the interface is in the correct state. Use the **show controllers** command followed by the type and address of the controller.

For complete descriptions of software configuration commands, refer to the publications listed in the "Related Documentation" section on page viii.

Performing a Basic Configuration

This section describes guidelines for performing a basic configuration: enabling the PA-VXA, PA-VXB, or PA-VXC port adapter, specifying the card type and controller, and entering various controller subcommands. After configuring the PA-VXA, PA-VXB, or PA-VXC port adapter in a Cisco 7200 series router, Cisco 7200 VXR router, Cisco 7301 router, Cisco 7401ASR router, or Cisco 7500 series router, see the "Configuring Voice over IP" section on page 4-6 for information on configuring your router for VoIP.

Specifying Card Type is Required

Because the port adapter can be configured for E1 or T1 connectivity, you must specify the card type as E1 or T1, as described in the following procedure. There is no default card type. The port adapter is not functional until the card type is set. Information about the port adapter is not indicated in the output of any show commands unless the card type has been set to E1 or T1.

Before using the **configure** command, you must enter the privileged level of the EXEC command interpreter with the **enable** command. The system prompts you for a password if one has been set.

Use the following procedure to configure the PA-VXA, PA-VXB, or PA-VXC port adapters. Press the **Return** key after each configuration step, unless otherwise noted.

Step 1 Confirm that the system recognizes the PA-VXA, PA-VXB, or PA-VXC port adapter by entering the **show running-config** command:

```
Router# show running-config
```

Step 2 Use the **configure terminal** command to enter configuration mode and specify that the console terminal is the source of the configuration subcommands:

```
Router# configure terminal
```

Step 3 Specify whether the card is to be used as T1 or E1 by using the **card type** command in configuration mode. The example below sets the card in slot 1 of a Cisco 7200 series router to T1:

```
Router(config)# card type t1 1
```

The example below sets the card in slot 1 of a Cisco 7200 series router to E1:

```
Router(config)# card type e1 1
```

The example below sets the card in port adapter slot 0 on a VIP in interface processor slot 1 of a Cisco 7500 series router to T1:

```
Router(config)# card type t1 1 0
```

The example below sets the card in port adapter slot 0 on a VIP in interface processor slot 1 of a Cisco 7500 series router to E1:

```
Router(config)# card type e1 1 0
```

Note To change the card type of the PA-VXA, PA-VXB, or PA-VXC after the **card type** command has been entered, you must remove the card from the router, save the running configuration to startup configuration, and reboot the router. When the router has finished rebooting, reinsert the card and repeat Step 3.

Step 4 Use the **controller** command in configuration mode to enter the controller configuration mode for the desired controller:

```
Router(config)# controller t1
Router(config-controller)#
```

Note The following steps must be done in controller configuration mode.

Step 5 Use the **framing** command to select carrier framing. If the card is set to T1, set framing to extended super frame (ESF) as shown in the example below:

```
Router(config-controller)# framing esf
```

If the card is set to E1, set framing to CRC4 as shown in the example below:

```
Router(config-controller)# framing crc4
```

Step 6 Use the **linecode** command to select the line coding. T1 cards should be set to binary 8-zero substitution (B8ZS) as shown in the example below:

```
Router(config-controller)# linecode b8zs
```

E1 cards should be set to HBD3 line code as shown in the example below:

```
Router(config-controller)# linecode hbd3
```

Step 7 Use the **clock source** command to select between internal or line clocking.

```
Router(config-controller)# clock source line
```

Note Line clocking is normally selected because of the highly accurate clock source supplied by the network.

Step 8 When installed in a Cisco 7200 VXR router, use the **frame-clock-select** *priority carrier-type controller* command in configuration mode to specify the clock source. This command may be used to specify backup clock sources, as shown in the example below:

```
Router(config)# frame-clock-select 1 T1 1/0
Router(config)# frame-clock-select 2 T1 1/1
```

The example above assigns T1 1/0 as the primary clock source. If that clock fails, T1 1/1 will become the primary clock source.

Step 9 Use the **ds0-group** *number* **timeslots** *range* **type** command to create a DS0 group.

```
Router(config-controller)# ds0-group 1 timeslots 10-18 type e&m-wink-start
```

Note The time slot range for a T1 card is 1 to 24, the time slot range for an E1 card is 1 to 30.

Step 10 Change the shutdown state to up and enable the interface:

```
Router(config-controller)# no shutdown
```

The **no shutdown** command passes an **enable** command to the PA-VXA, PA-VXB, or PA-VXC port adapter. It also causes the PA-VXA, PA-VXB, or PA-VXC port adapter to configure itself based on the previous configuration commands sent.

Configuring Voice over IP

Voice Over IP (VoIP) enables a Cisco 7200 series router, Cisco 7200 VXR router, Cisco 7301 router, Cisco 7401ASR router, or Cisco 7500 series router to carry voice traffic (for example, telephone calls and faxes) over an IP network.

VoIP offers the following benefits:

- Toll bypass
- Remote PBX presence over WANs
- Unified voice and data trunking
- POTS-Internet telephony gateways

Prerequisite Tasks

Before you can configure your Cisco 7200 series router, Cisco 7200 VXR router, Cisco 7301 router, Cisco 7401ASR router, or Cisco 7500 series router to use VoIP, you must first:

- Establish a working IP network. For more information about configuring IP, refer to the "IP Overview" and "IP Addressing and Services" chapters in the Cisco IOS Release 12.0 *Network Protocols Configuration Guide, Part 1*.
- Install the PA-VXA, PA-VXB, or PA-VXC port adapter in your router.
- Complete your company's dial plan.
- Establish a working telephony network based on your company's dial plan.
- Integrate your dial plan and telephony network into your existing IP network topology. Merging your IP and telephony networks depends on your particular IP and telephony network topology. In general, we make the following suggestions:
  - Use canonical numbers wherever possible. It is important to avoid situations where numbering systems are significantly different on different routers or access servers in your network.
  - Make routing or dialing transparent to the user—for example, avoid secondary dial tones from secondary switches, where possible.
  - Contact your PBX vendor for instructions about how to reconfigure the appropriate PBX interfaces.

After you have analyzed your dial plan and decided how to integrate it into your existing IP network, you are ready to configure your network devices to support VoIP.

How VoIP Handles a Typical Telephone Call

Before configuring VoIP on your Cisco 7200 series router, Cisco 7200 VXR router, Cisco 7301 router, Cisco 7401ASR router, or Cisco 7500 series router, it helps to understand what happens at an application level when you place a call using VoIP. The general flow of a two-party voice call using VoIP is as follows:

1. The user picks up the handset; this signals an off-hook condition to the signaling application part of VoIP in the Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7301 router, Cisco 7401ASR routers, or Cisco 7500 series routers.
2. The session application part of VoIP issues a dial tone and waits for the user to dial a telephone number.
3. The user dials the telephone number; those numbers are accumulated and stored by the session application.
4. After enough digits are accumulated to match a configured destination pattern, the telephone number is mapped to an IP host through the dial plan mapper. The IP host has a direct connection to either the destination telephone number or a PBX that is responsible for completing the call to the configured destination pattern.
5. The session application then runs the H.323 session protocol to establish a transmission and a reception channel for each direction over the IP network. If the call is being handled by a PBX, the PBX forwards the call to the destination telephone. If Resource Reservation Protocol (RSVP) has been configured, the RSVP reservations are put into effect to achieve the desired quality of service over the IP network.
6. The codecs are enabled for both ends of the connection and the conversation proceeds using Realtime Transport Protocol/User Datagram Protocol/Internet Protocol (RTP/UDP/IP) as the protocol stack.
7. Any call-progress indications (or other signals that can be carried in-band) are cut through the voice path as soon as an end-to-end audio channel is established. Signaling that can be detected by the voice ports (for example, in-band DTMF digits after the call setup is complete) is also trapped by the session application at either end of the connection and carried over the IP network encapsulated in Real Time Conferencing Protocol (RTCP) using the RTCP extension mechanism.
8. When either end of the call hangs up, the RSVP reservations are torn down (if RSVP is used) and the session ends. Each end becomes idle, waiting for the next off-hook condition to trigger another call setup.

Configuration Tasks

To configure VoIP on the Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7301 router, Cisco 7401ASR routers, or Cisco 7500 series routers, you need to perform the following steps:

Step 1 Configure your IP network to support real-time voice traffic. Fine-tuning your network to adequately support VoIP involves a series of protocols and features geared toward quality of service (QoS). To configure your IP network for real-time voice traffic, you need to take into consideration the entire scope of your network, and then select and configure the appropriate QoS tool or tools:

- RSVP
- Multilink PPP with interleaving
- RTP header compression

- Custom queuing
- Weighted fair queuing

See the "Configure IP Networks for Real-Time Voice Traffic" section on page 4-9 for information about how to select and configure the appropriate QoS tools to optimize voice traffic on your network.

Step 2 (Optional) If you plan to run VoIP over Frame Relay, you need to take certain factors into consideration when configuring VoIP for it to run smoothly over Frame Relay. For example, a public Frame Relay cloud provides no guarantees for QoS. See the "Configuring Voice over Frame Relay" section on page 4-32 for information about deploying VoIP over Frame Relay.

Step 3 Use the **num-exp** command to configure number expansion if your telephone network is configured so that you can reach a destination by dialing only a portion (an extension number) of the full E.164 telephone number. See the "Configure Number Expansion" section on page 4-14 for information about number expansion.

Step 4 Use the **dial-peer voice** command to define dial peers and switch to the dial-peer configuration mode. Each dial peer defines the characteristics associated with a call leg. A call leg is a discrete segment of a call connection that lies between two points in the connection. An end-to-end call comprises four call legs, two from the perspective of the source router, and two from the perspective of the destination router. Dial peers are used to apply attributes to call legs and to identify call origin and destination. There are two different kinds of dial peers:

- POTS—Dial peer describing the characteristics of a traditional telephony network connection. POTS peers point to a particular voice port on a voice network device. To minimally configure a POTS dial peer, you need to configure the following two characteristics: associated telephone number and logical interface. Use the **destination-pattern** command to associate a telephone number with a POTS peer. Use the **port** command to associate a specific logical interface with a POTS peer. In addition, you can specify direct inward dialing for a POTS peer by using the **direct-inward-dial** command.
- VoIP—Dial peer describing the characteristics of a packet network connection; in the case of VoIP, this is an IP network. VoIP peers point to specific VoIP devices. To minimally configure a VoIP peer, you need to configure the following two characteristics: associated destination telephone number and a destination IP address. Use the **destination-pattern** command to define the destination telephone number associated with a VoIP peer. Use the **session-target** command to specify a destination IP address for a VoIP peer.

In addition, you can use VoIP peers to define characteristics such as IP precedence, additional QoS parameters (when RSVP is configured), codec, and voice activation detection (VAD). Use the **ip precedence** command to define IP precedence. If you have configured RSVP, use either the **req-qos** or **acc-qos** command to configure QoS parameters. Use the **codec** command to configure specific voice coder rates. Use the **vad** command to disable voice activation detection and the transmission of silence packets.

See the "Configure Dial Peers" section on page 4-16 and the "Optimize Dial Peer and Network Interface Configurations" section on page 4-29 for additional information about configuring dial peers and dial-peer characteristics.

Step 5 You need to configure your router to support voice ports. In general, voice-port commands define the characteristics associated with a particular voice-port signaling type. Voice ports on the Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7301 router, Cisco 7401ASR routers, and Cisco 7500 series routers support three basic voice signaling types:

- FXO—Foreign Exchange Office interface
- FXS—Foreign Exchange Station interface
- E&M—"RecEive and TransMit" interface or the "Ear and Mouth" interface

Under most circumstances, the default voice-port command values are adequate to configure FXO and FXS ports to transport voice data over your existing IP network. Because of the inherent complexities involved with PBX networks, E&M ports might need specific voice-port values configured, depending on the specifications of the devices in your telephony network. For information about configuring voice ports, see the "Configure Voice Ports" section on page 4-22.

Configure IP Networks for Real-Time Voice Traffic

You need to have a well-engineered end-to-end network when you run delay-sensitive applications such as VoIP. Fine-tuning your network to adequately support VoIP involves a series of protocols and features geared toward quality of service (QoS). It is beyond the scope of this document to explain the specific details relating to wide-scale QoS deployment. Cisco IOS software provides many tools for enabling QoS on your backbone, such as random early detection (RED), weighted random early detection (WRED), fancy queuing (meaning custom, priority, or weighted fair queuing), and IP precedence. To configure your IP network for real-time voice traffic, you must consider the entire scope of your network, and then select the appropriate QoS tool or tools.

The important thing to remember is that QoS must be configured throughout your network—not just on the Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7401ASR routers, or Cisco 7500 series routers running VoIP—to improve voice network performance. Not all QoS techniques are appropriate for all network routers. Edge routers and backbone routers in your network do not necessarily perform the same operations; the QoS tasks they perform might differ as well. To configure your IP network for real-time voice traffic, you need to take into consideration the functions of both edge and backbone routers in your network, and then select the appropriate QoS tool or tools.

In general, edge routers perform the following QoS functions:

- Packet classification
- Admission control
- Bandwidth management
- Queuing

In general, backbone routers perform the following QoS functions:

- High-speed switching and transport
- Congestion management
- Queue management

Scalable QoS solutions require cooperative edge and backbone functions.

Although not mandatory, some QoS tools have been identified as valuable in fine-tuning your network to support real-time voice traffic. To configure your IP network for QoS using these tools, perform one or more of the following tasks:

- Configure RSVP for Voice, page 4-10
- Configure Multilink PPP with Interleaving, page 4-11
- Configure RTP Header Compression, page 4-12
- Configure Custom Queuing, page 4-14
- Configure Weighted Fair Queuing, page 4-14

Each of these tasks is discussed in the following sections.

Configure RSVP for Voice

RSVP allows end systems to request a particular quality of service (QoS) from the network. Real-time voice traffic requires network consistency. Without consistent QoS, real-time traffic can experience jitter, insufficient bandwidth, delay variations, or information loss. RSVP works in conjunction with current queuing mechanisms. It is up to the interface queuing mechanism (such as weighted fair queuing or weighted random early detection) to implement the reservation.

RSVP can be equated to a dynamic access list for packet flows.

You should configure RSVP to ensure QoS if the following conditions exist in your network:

- Small-scale voice network implementation
- Slow links
- Links with high utilization
- Links less than 2 Mbps
- Need for the best possible voice quality

Enable RSVP

To minimally configure RSVP for voice traffic, you must enable RSVP on each interface where priority needs to be set.

By default, RSVP is disabled so that it is backward compatible with systems that do not implement RSVP. To enable RSVP on an interface, use the following command in interface configuration mode:

ip rsvp bandwidth [*interface-kbps*] [*single-flow-kbps*]

This command starts RSVP and sets the bandwidth and single-flow limits. The default maximum bandwidth is up to 75 percent of the bandwidth available on the interface. By default, the amount reservable by a flow can be up to the entire reservable bandwidth.

On subinterfaces, this command applies the more restrictive of the available bandwidths of the physical interface and the subinterface.

Reservations on individual circuits that do not exceed the single-flow limit normally succeed. If, however, reservations have been made on other circuits adding up to the line speed, and a reservation is made on a subinterface which itself has enough remaining bandwidth, it will still be refused because the physical interface lacks supporting bandwidth.

Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7301 router, Cisco 7401ASR routers, and Cisco 7500 series routers running VoIP and configured for RSVP request allocations according to the following formula:

bps = packet_size + ip/udp/rtp header size * 50 per second

For G.729, the allocation is 24,000 bps. For G.711, the allocation is 80,000 bps.

RSVP Configuration Example

The following example enables RSVP and sets the maximum bandwidth to 100 kbps and the maximum bandwidth per single request to 32 kbps (the example presumes that both VoIP dial peers have been configured):

```
interface DSPfarm 1/0/0
 ip rsvp bandwidth 100 32
 fair-queue
 end
!
dial-peer voice 1211 voip
 req-qos controlled-load
!
dial-peer voice 1212 voip
 req-qos controlled-load
```

Configure Multilink PPP with Interleaving

Multiclass Multilink PPP interleaving allows large packets to be multilink-encapsulated and fragmented into smaller packets to satisfy the delay requirements of real-time voice traffic; small real-time packets, which are not multilink-encapsulated, are transmitted between fragments of the large packets. The interleaving feature also provides a special transmit queue for the smaller, delay-sensitive packets, enabling them to be transmitted earlier than other flows. Interleaving provides the delay bounds for delay-sensitive voice packets on a slow link that is used for other best-effort traffic.

Note Interleaving applies only to interfaces that can configure a multilink bundle interface. These include virtual templates, dialer interfaces, and Integrated Services Digital Network (ISDN) Basic Rate Interface (BRI) or Primary Rate Interface (PRI).

In general, Multilink PPP with interleaving is used in conjunction with weighted fair queuing and RSVP or IP precedence to ensure voice packet delivery. Use Multilink PPP with interleaving and weighted fair queuing to define how data will be managed; use RSVP or IP precedence to give priority to voice packets.

You should configure Multilink PPP if the following conditions exist in your network:

- Point-to-point connection using PPP encapsulation
- Slow links

Note Do not use Multilink PPP on links greater than 2 Mbps.

Enable Multilink PPP with Interleaving

Multilink PPP support for interleaving can be configured on virtual templates, dialer interfaces, and ISDN BRI or PRI interfaces. To configure interleaving, you need to complete the following tasks:

- Configure the dialer interface or virtual template, as defined in the relevant chapters of the Cisco IOS Release 12.0 *Dial Solutions Configuration Guide*.
- Configure Multilink PPP and interleaving on the interface or template.

To configure Multilink PPP and interleaving on a configured and operational interface or virtual interface template, use the following commands in interface mode:

	Command	Purpose
Step 1	**ppp multilink**	Enable Multilink PPP.
Step 2	**ppp multilink interleave**	Enable real-time packet interleaving.
Step 3	**ppp multilink fragment-delay** *milliseconds*	Optionally, configure a maximum fragment delay.
Step 4	**ip rtp reserve** *lowest-UDP-port range-of-ports* [*maximum-bandwidth*]	Reserve a special queue for real-time packet flows to specified destination User Datagram Protocol (UDP) ports, allowing real-time traffic to have higher priority than other flows. This is only applicable if you have not configured RSVP.

Note The **ip rtp reserve** command can be used instead of configuring RSVP. If you configure RSVP, this command is not required.

For more information about Multilink PPP, refer to the "Configuring Media-Independent PPP and Multilink PPP" chapter in the Cisco IOS Release 12.0 *Dial Solutions Configuration Guide*.

Multilink PPP Configuration Example

The following example defines a virtual interface template that enables Multilink PPP with interleaving and a maximum real-time traffic delay of 20 milliseconds, and then applies that virtual template to the Multilink PPP bundle:

```
interface virtual-template 1
 ppp multilink
 encapsulated ppp
 ppp multilink interleave
 ppp multilink fragment-delay 20
 ip rtp reserve 16384 100 64

multilink virtual-template 1
```

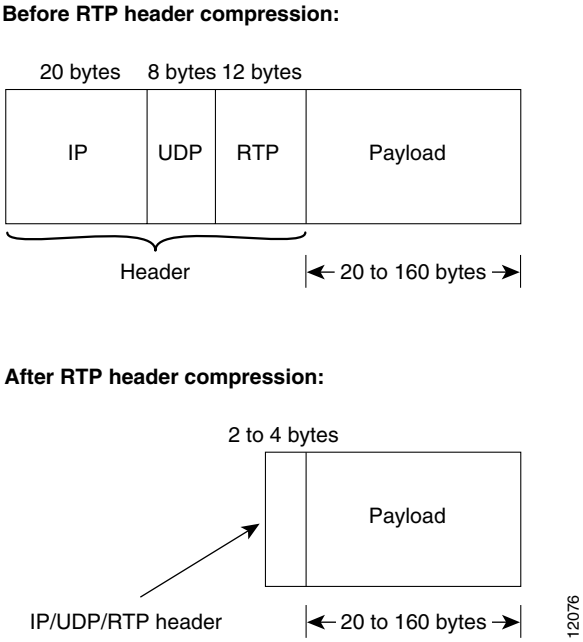
Configure RTP Header Compression

Real-Time Transport Protocol (RTP) is used for carrying packetized audio traffic over an IP network. RTP header compression compresses the IP/UDP/RTP header in an RTP data packet from 40 bytes to approximately 2 to 4 bytes (most of the time), as shown in Figure 4-1.

This compression feature is beneficial if you are running VoIP over slow links. Enabling compression on both ends of a low-bandwidth serial link can greatly reduce the network overhead if there is a lot of RTP traffic on that slow link.

Typically, an RTP packet has a payload of approximately 20 to 160 bytes for audio applications that use compressed payloads. RTP header compression is especially beneficial when the RTP payload size is small (for example, compressed audio payloads between 20 and 50 bytes).

Figure 4-1 RTP Header Compression



You should configure RTP header compression if the following conditions exist in your network:

- Slow links
- Need to save bandwidth

Note Do not use RTP header compression on links greater than 2 Mbps.

Perform the following tasks to configure RTP header compression for VoIP. The first task is required; the second task is optional.

- Enable RTP Header Compression on a Serial Interface, page 4-13
- Change the Number of Header Compression Connections, page 4-14

Enable RTP Header Compression on a Serial Interface

To use RTP header compression, you need to enable compression on both ends of a serial connection. To enable RTP header compression, use the following command in interface configuration mode:

```
ip rtp header-compression [passive]
```

If you include the **passive** keyword, the software compresses outgoing RTP packets only if incoming RTP packets on the same interface are compressed. If you use the command without the **passive** keyword, the software compresses all RTP traffic.

Change the Number of Header Compression Connections

By default, the software supports a total of 16 RTP header compression connections on an interface. To specify a different number of RTP header compression connections, use the following command in interface configuration mode:

ip rtp compression-connections *number*

RTP Header Compression Configuration Example

The following example enables RTP header compression for a serial interface:

```
interface serial 0
 ip rtp header-compression
 encapsulation ppp
 ip rtp compression-connections 25
```

Configure Custom Queuing

Some QoS features, such as IP RTP reserve and custom queuing, are based on the transport protocol and the associated port number. Real-time voice traffic is carried on UDP ports ranging from 16384 to 16624. This number is derived from the following formula:

16384 = 4 multiplied by the number of voice ports in the Cisco 7200 VXR router

Custom queuing and other methods for identifying high-priority streams should be configured for these port ranges.

Configure Weighted Fair Queuing

Weighted fair queuing ensures that queues do not starve for bandwidth and that traffic gets predictable service. Low-volume traffic streams receive preferential service; high-volume traffic streams share the remaining capacity, obtaining equal or proportional bandwidth.

In general, weighted fair queuing is used in conjunction with Multilink PPP with interleaving and RSVP or IP precedence to ensure voice packet delivery. Use weighted fair queuing with Multilink PPP to define how data will be managed; use RSVP or IP precedence to give priority to voice packets.

Configure Number Expansion

In most corporate environments, the telephone network is configured so that you can reach a destination by dialing only a portion (an extension number) of the full E.164 telephone number. VoIP can be configured to recognize extension numbers and expand them into their full E.164 dialed numbers by using two commands in tandem: **destination-pattern** and **num-exp**. Before you configure these two commands, it is helpful to map individual telephone extensions with their full E.164 dialed numbers. This can be done easily by creating a number expansion table.

Create a Number Expansion Table

In Figure 4-2, a small company wants to use VoIP to integrate its telephony network with its existing IP network. The destination pattern (or expanded telephone number) associated with Router A (located to the left of the IP cloud) is (408) 555-*xxxx*, where *xxxx* identifies the individual dial peers by extension. The destination pattern (or expanded telephone number) associated with Router B (located to the right of the IP cloud) is (729) 411-*xxxx*.

Figure 4-2 Sample VoIP Network

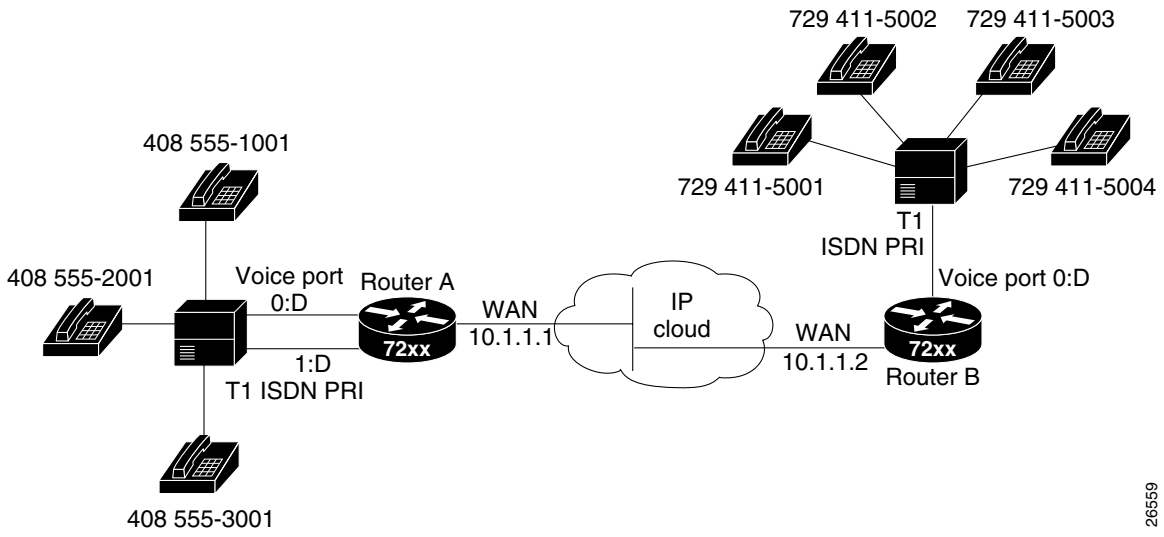


Table 4-2 shows the number expansion table for this scenario.

Table 4-2 Sample Number Expansion Table

Extension	Destination Pattern	num-exp Command Entry
1001	4085551001	**num-exp 1001 4085551001**
2001	4085552001	**num-exp 2001 4085552001**
3....	4085553...	**num-exp 3... 4085553...**
5001	7294115001	**num-exp 5001 7294115001**
5002	7294115002	**num-exp 5002 7294115002**
5...	7294115...	**num-exp 5001 7294115...**

Note You can use the period symbol (.) to represent variables (such as extension numbers) in a telephone number.

The information included in this example needs to be configured on both Router A and Router B.

Expand a Number

To define how to expand an extension number into a particular destination pattern, use the following command in global configuration mode:

```
num-exp extension-number extension-string
```

You can verify the number expansion information by using the **show num-exp** command to verify that you have mapped the telephone numbers correctly.

After you have configured dial peers and assigned destination patterns to them, you can verify number expansion information by using the **show dialplan number** command to see how a telephone number maps to a dial peer.

Configure Dial Peers

The key point to understanding how VoIP functions is to understand dial peers. Each dial peer defines the characteristics associated with a call leg, as shown in Figure 4-3 and Figure 4-4. A call leg is a discrete segment of a call connection that lies between two points in the connection. All of the call legs for a particular connection have the same connection ID.

There are two different kinds of dial peers:

- POTS—Dial peer describing the characteristics of a traditional telephony network connection. POTS peers point to a particular voice port on a voice network device.
- VoIP—Dial peer describing the characteristics of a packet network connection; in the case of VoIP, this is an IP network. VoIP peers point to specific VoIP devices.

An end-to-end call comprises four call legs, two from the perspective of the source router as shown in Figure 4-3, and two from the perspective of the destination router as shown in Figure 4-4. A dial peer is associated with each one of these call legs. Dial peers are used to apply attributes to call legs and to identify call origin and destination. Attributes applied to a call leg include QoS, codec, VAD, and fax rate.

Figure 4-3 Dial Peer Call Legs from the Perspective of the Source Router

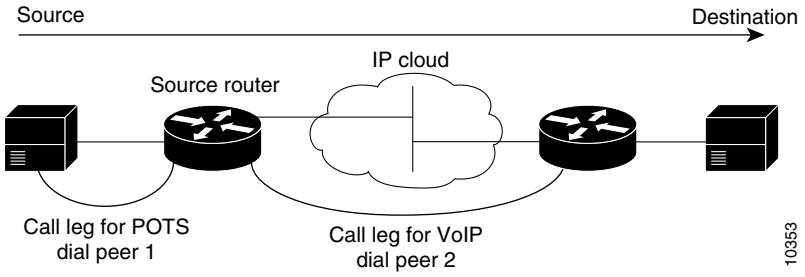
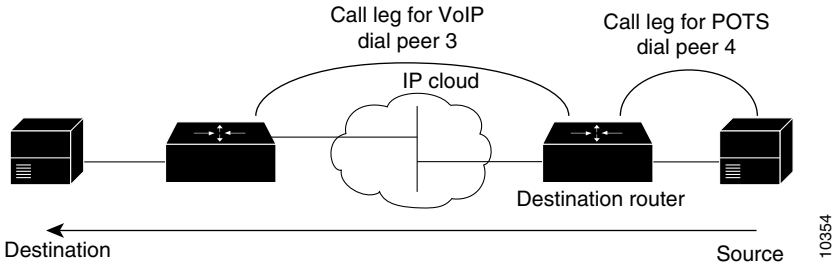


Figure 4-4 Dial Peer Call Legs from the Perspective of the Destination Router



Inbound Versus Outbound Dial Peers

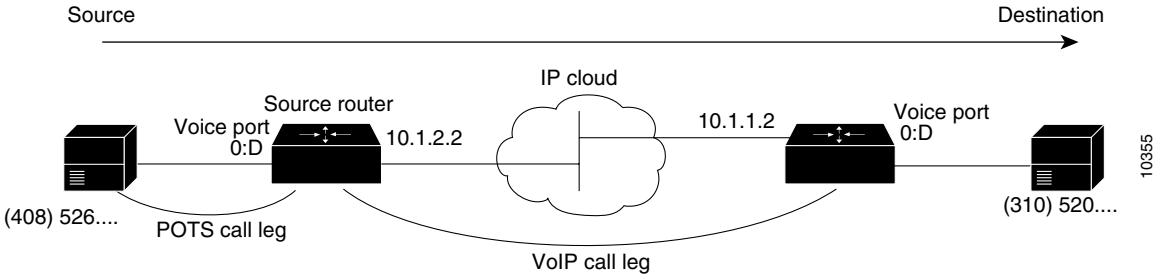
Dial peers are used for both inbound and outbound call legs. It is important to remember that these terms are defined from the *router's* perspective. An inbound call leg originates *outside* the router. An outbound call leg originates *from* the router.

For inbound call legs, a dial peer might be associated with the calling number or the port designation. Outbound call legs always have a dial peer associated with them. The destination pattern is used to identify the outbound dial peer. The call is associated with the outbound dial peer at setup time.

POTS peers associate a telephone number with a particular voice port so that incoming calls for that telephone number can be received and outgoing calls can be placed. VoIP peers point to specific devices (by associating destination telephone numbers with a specific IP address) so that incoming calls can be received and outgoing calls can be placed. Both POTS and VoIP peers are needed to establish VoIP connections.

Establishing communication using VoIP is similar to configuring an IP static route: you are establishing a specific voice connection between two defined endpoints. As shown in Figure 4-5, for outgoing calls (from the perspective of the POTS dial peer 1), the POTS dial peer establishes the source (through the originating telephone number or voice port) of the call. The VoIP dial peer establishes the destination by associating the destination telephone number with a specific IP address.

Figure 4-5 Outgoing Calls from the Perspective of POTS Dial Peer 1



To configure call connectivity between the source and destination as illustrated in Figure 4-5, enter the following commands on router 10.1.2.2:

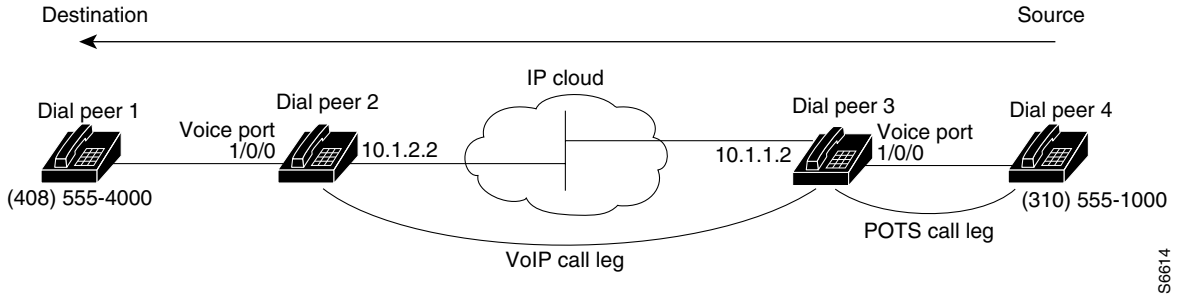
```
dial-peer voice 1 pots
 destination-pattern 1408526....
 port 1/0/0

dial-peer voice 2 voip
 destination-pattern 1310520....
 session target ipv4:10.1.1.2
```

In the previous configuration example, the last four digits in the VoIP dial peer's destination pattern were replaced with wildcards. This means that from router 10.1.2.2, calling any number string that begins with the digits "1310520" results in a connection to router 10.1.1.2. This implies that router 10.1.1.2 services all numbers beginning with those digits. From router 10.1.1.2, calling any number string that begins with the digits "1408526" will result in a connection to router 10.1.2.2. This implies that router 10.1.2.2 services all numbers beginning with those digits. For more information about stripping and adding digits, see the "Outbound Dialing on POTS Peers" section on page 4-19.

Figure 4-6 shows how to complete the end-to-end call between dial peer 1 and dial peer 4.

Figure 4-6 Outgoing Calls from the Perspective of POTS Dial Peer 2



To complete the end-to-end call between dial peer 1 and dial peer 4 as illustrated in Figure 4-6, enter the following commands on router 10.1.1.2:

```
dial-peer voice 4 pots
 destination-pattern 1310555....
 port 1/0/0

dial-peer voice 3 voip
 destination-pattern 1408555....
 session target ipv4:10.1.2.2
```

Create a Peer Configuration Table

There is specific data relative to each dial peer that needs to be identified before you can configure dial peers in VoIP. One way to do this is to create a peer configuration table.

Using the example in Figure 4-2, Router A, with an IP address of 10.1.1.1, connects a small sales branch office to the main office through Router B. There are three telephones in the sales branch office that need to be established as dial peers. Router B, with an IP address of 10.1.1.2, is the primary gateway to the main office; as such, it needs to be connected to the company's PBX. There are four devices that need to be established as dial peers in the main office, all of which are basic telephones connected to the PBX. Figure 4-2 shows a diagram of this small voice network.

Table 4-3 shows the peer configuration table for the example illustrated in Figure 4-2.

Configure POTS Peers

Table 4-3 Peer Configuration Table for Sample VoIP Network

Dial Peer Tag	Ext	Dest-Pattern	Type	Session-Target	Codec	QoS
Router A						
1	6....	+1408555....	POTS			
10		+1729411....	VoIP	IPV4 10.1.1.2	G.729	Best Effort
Router B						
11		+1408555....	VoIP	IPV4 10.1.1.1	G.729	Best Effort
4	2....	+1729411....	POTS			

POTS peers enable incoming calls to be received by a particular telephony device. To configure a POTS peer, you need to uniquely identify the peer (by assigning it a unique tag number), define its telephone numbers, and associate it with a voice port through which calls will be established. Under most circumstances, the default values for the remaining dial-peer configuration commands will be sufficient to establish connections.

To enter the dial-peer configuration mode (and select POTS as the method of voice-related encapsulation), use the following command in global configuration mode:

```
dial-peer voice number pots
```

The *number* value of the **dial-peer voice pots** command is a tag that uniquely identifies the dial peer. (This number has local significance only.)

To configure the identified POTS peer, use the following commands in dial-peer configuration mode:

	Command	Purpose
Step 1	**destination-pattern** ***string***	Define the destination telephone number associated with this POTS dial peer.
Step 2	**port** ***controller number***	Associate this POTS dial peer with a specific logical dial interface.

Outbound Dialing on POTS Peers

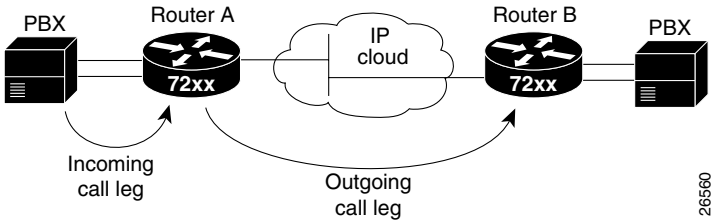
When a router receives a voice call, it selects an outbound dial peer by comparing the called number (the full E.164 telephone number) in the call information with the number configured as the destination pattern for the POTS peer. The router then strips out the left-justified numbers corresponding to the destination pattern matching the called number. If you have configured a prefix, the prefix will be put in front of the remaining numbers, creating a dial string, which the router will then dial. If all numbers in the destination pattern are stripped out, the user receives (depending on the attached equipment) a dial tone.

For example, suppose there is a voice call for which the E.164 called number is 1 (310) 767-2222. If you configure a destination pattern of “1310767” and a prefix of “9,” the router strips out “1310767” from the E.164 telephone number, leaving the extension number of “2222.” It then appends the prefix “9” to the front of the remaining numbers, so that the actual numbers dialed are “9, 2222.” The comma in this example means that the router will pause for one second between dialing the “9” and the “2” to allow for a secondary dial tone.

Direct Inward Dial for POTS Peers

Direct inward dial (DID) is used to determine how the called number is treated for incoming POTS call legs. As shown in Figure 4-7, incoming means from the perspective of the router. In this case, it is the call leg coming into the router to be forwarded through to the appropriate destination pattern.

Figure 4-7 Incoming and Outgoing POTS Call Legs



Unless otherwise configured, when a call arrives on the router, the router presents a dial tone to the caller and collects digits until it can identify the destination dial peer. After the dial peer has been identified, the call is forwarded through the next call leg to the destination.

There are cases when it might be necessary for the router to use the called number Dialed Number Identification Service (DNIS) to find a dial peer for the outgoing call leg—for example, if the switch connecting the call to the router has already collected the digits. DID enables the router to match the called number with a dial peer and then directly place the outbound call. With DID, the router does not present a dial tone to the caller and does not collect digits; it forwards the call directly to the configured destination.

To use DID and an incoming called number, a dial peer must be associated with the incoming call leg. It is helpful to understand the logic behind the algorithm used to associate the incoming call leg with the dial peer.

The algorithm used to associate incoming call legs with dial peers uses three inputs (which are derived from signaling and interface information associated with the call) and four defined dial peer elements. The three signaling inputs are:

- Called number (DNIS)—Set of numbers representing the destination, which is derived from the ISDN setup message or CAS DNIS.
- Calling number (ANI)—Set of numbers representing the origin, which is derived from the ISDN setup message or CAS DNIS.
- Voice port—Voice port carrying the call.

The four defined dial peer elements are:

- Destination pattern—Pattern representing the phone numbers to which the peer can connect.
- Answer address—Pattern representing the phone numbers from which the peer can connect.
- Incoming called number—Pattern representing the phone numbers that associate an incoming call leg to a peer based on the called number or DNIS.
- Port—Port through which calls to this peer are placed.

Using the elements, the algorithm is as follows:

For all peers where call type (VoIP versus POTS) matches dial peer type:

if the type is matched, associate the called number with the incoming called-number

else if the type is matched, associate calling-number with answer-address

else if the type is matched, associate calling-number with destination-pattern

else if the type is matched, associate voice port to port

This algorithm shows that if a value is not configured for answer address, the origin address is used because, in most cases, the origin address and answer address are the same.

To configure DID for a particular POTS dial peer, use the following commands, initially in global configuration mode:

	Command	Purpose
Step 1	**dial-peer voice** *number* **pots**	Enter the dial-peer configuration mode to configure a POTS peer.
Step 2	**direct-inward-dial**	Specify direct inward dial for this POTS peer.

Note Direct inward dial is configured for the calling POTS dial peer.

Configure VoIP Peers

VoIP peers enable outgoing calls to be made from a particular telephony device. To configure a VoIP peer, you need to uniquely identify the peer (by assigning it a unique tag number) and define its destination telephone number and destination IP address. As with POTS peers, under most circumstances, the default values for the remaining dial-peer configuration commands are adequate to establish connections.

To enter the dial-peer configuration mode (and select VoIP as the method of voice-related encapsulation), use the following command in global configuration mode:

dial-peer voice *number* **voip**

The *number* value of the **dial-peer voice voip** command is a tag that uniquely identifies the dial peer.

To configure the identified VoIP peer, use the following commands in dial-peer configuration mode:

	Command	Purpose
Step 1	**destination-pattern** *string*	Define the destination telephone number associated with this VoIP dial peer.
Step 2	**session-target** {**ipv4:***destination-address* | **dns:***host-name*}	Specify a destination IP address for this dial peer.

Verify Configuration

You can check the validity of your dial-peer configuration by performing the following tasks:

- If you have relatively few dial peers configured, you can use the **show dial-peer voice** command to verify that the data configured is correct. Use this command to display a specific dial peer or to display all configured dial peers.
- Use the **show dialplan number** command to show the dial peer to which a particular number (destination pattern) resolves.

Troubleshooting Tips

If you are having trouble connecting a call and you suspect the problem is associated with dial-peer configuration, you can try to resolve the problem by performing the following tasks:

- Ping the associated IP address to confirm connectivity.
- Use the **show dial-peer voice** command to verify that the operational status of the dial peer is up.
- Use the **show dialplan number** command on the local and remote routers to verify that the data is configured correctly on both.
- If you have configured number expansion, use the **show num-exp** command to check that the partial number on the local router maps to the correct full E.164 telephone number on the remote router.
- If you have configured a codec value, there can be a problem if both VoIP dial peers on either side of the connection have incompatible codec values. Make sure that both VoIP peers have been configured with the same codec value.
- Use the **debug vpm spi** command to verify the output string that the router dials is correct.
- Use the **debug cch323 rtp** command to check RTP packet transport.
- Use the **debug cch323 h225** command to check the call setup.

Configure Voice Ports

Voice ports on Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7301 router, Cisco 7401ASR routers, and Cisco 7500 series routers support three basic voice signaling types:

- FXO—The Foreign Exchange Office interface allows a connection to be directed to the PSTN's central office (or to a standard PBX interface, if the local telecommunications authority permits). This interface is useful for off-premises extension applications.
- FXS—The Foreign Exchange Station interface allows connection for basic telephone equipment, key sets, and PBXs, and supplies ring, voltage, and dial tone.
- E&M—The "Ear and Mouth" interface (or "RecEive and TransMit" interface) allows connection for PBX trunk lines (tie lines). It is a signaling technique for two-wire and four-wire telephone and trunk interfaces.

In general, voice-port commands define the characteristics associated with a particular voice-port signaling type. Under most circumstances, the default voice-port command values are adequate to configure FXO and FXS ports to transport voice data over your existing IP network. Because of the inherent complexities involved with PBX networks, E&M ports might need specific voice-port values configured, depending on the specifications of the devices in your telephony network.

Configure FXO or FXS Voice Ports

Under most circumstances, the default voice-port values are adequate for both FXO and FXS voice ports. If you need to change the default configuration for these voice ports, perform the following tasks. Items included in Step 1 and Step 2 are required; items included in Step 3 are optional.

Step 1 Identify the voice port and enter the voice-port configuration mode by using the **voice-port** command.

Step 2 Configure the following mandatory voice-port parameters by using the indicated commands:

- Dial type (FXO only) using the **dial-type** command
- Signal type using the **signal** command
- Call progress tone using the **cptone** command
- Ring frequency (FXS only) using the **ring frequency** command
- Ring number (FXO only) using the **ring number** command

Step 3 Configure one or more of the following optional voice-port parameters by using the indicated commands:

- PLAR connection mode using the **connection plar** command
- Music threshold using the **music-threshold** command
- Description using the **description** command
- Comfort noise (if VAD is activated—**vad** is a dial-peer command) using the **comfort-noise** command

To configure FXO and FXS voice ports, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	**configure terminal**	Enter the global configuration mode.
Step 2	**voice-port** *slot-number/subunit-number/port*	Identify the voice port you want to configure and enter the voice-port configuration mode.
Step 3	**dial-type** {**dtmf** | **pulse**}	(For FXO ports only.) Select the appropriate dial type for out-dialing.
Step 4	**signal** {**loop-start** | **ground-start**}	Select the appropriate signal type for this interface.
Step 5	**cptone** *country*	Select the appropriate voice call progress tone for this interface. The default for this command is **us**. For a list of supported countries, refer to the *Voice, Video, and Home Applications Command Reference*.
Step 6	**ring number** *number*	(For FXO ports only.) Specify the maximum number of rings to be detected before answering a call.
Step 7	**connection plar** *string*	(Optional.) Specify the Private Line Auto Ringdown (PLAR) connection if this voice port is used for a PLAR connection. The *string* value specifies the destination telephone number.

	Command	Purpose
Step 8	**music-threshold** *number*	(Optional.) Specify the threshold (in decibels) for on-hold music. Valid entries are from –70 dB to –30 dB.
Step 9	**description** *string*	(Optional.) Attach descriptive text about this voice-port connection.
Step 10	**comfort-noise**	(Optional.) Specify that background noise will be generated.

Validation Tips

You can check the validity of your voice-port configuration by performing the following tasks:

- Pick up the handset of an attached telephony device and check for a dial tone.
- If you have dial tone, check for dual tone multifrequency (DTMF) detection. If the dial tone stops when you dial a digit, then the voice port is most likely configured properly.
- Use the **show voice-port** command to verify that the data configured is correct.

Troubleshooting Tips

If you are having trouble connecting a call and you suspect the problem is associated with voice-port configuration, you can try to resolve the problem by performing the following tasks:

- Ping the associated IP address to confirm connectivity.
- Use the **show voice-port** command to make sure that the port is enabled. If the port is offline, use the **no shutdown** command.
- If you have configured E&M interfaces, make sure that the values pertaining to your specific PBX setup, such as timing or type, are correct.
- Check to see if the voice network module has been correctly installed. For more information, refer to the installation document *Voice Network Module and Voice Interface Card Configuration Note* that came with your voice network module.

Fine-Tune FXO and FXS Voice Ports

Depending on the specifics of your particular network, you might need to adjust voice parameters involving timing, input gain, and output attenuation for FXO or FXS voice ports. Collectively, these commands are referred to as voice-port tuning commands.

Note In most cases, the default values for voice-port tuning commands are sufficient.

To configure voice-port tuning for FXO and FXS voice ports, perform the following steps:

Step 1 Identify the voice port and enter the voice-port configuration mode using the **voice-port** command.

Step 2 For each of the following parameters, select the appropriate value using the commands indicated:

- Input gain using the **input gain** command
- Output attenuation using the **output attenuation** command
- Echo cancel coverage using the **echo-cancel enable** and **echo-cancel coverage** commands

- Nonlinear processing using the **non-linear** command
- Initial digit timeouts using the **timeouts initial** command
- Interdigit timeouts using the **timeouts interdigits** command
- Timing other than timeouts using the **timing digit**, **timing inter-digit**, **timing pulse-digit**, and **timing pulse-inter-digit** commands

To fine-tune FXO or FXS voice ports, use the following commands beginning in privileged EXEC mode:

:

	Command	Purpose
Step 1	**configure terminal**	Enter the global configuration mode.
Step 2	**voice-port** *slot-number/subunit-number/port*	Identify the voice port you want to configure and enter the voice-port configuration mode.
Step 3	**echo-cancel enable**	Enable echo cancellation of voice that is sent out the interface and received back on the same interface.
Step 4	**echo-cancel coverage** *value*	Adjust the size (in milliseconds) of the echo cancel. Acceptable values are 16, 24, and 32.
Step 5	**non-linear**	Enable nonlinear processing, which shuts off any signal if no near-end speech is detected. (Nonlinear processing is used with echo cancellation.)
Step 6	**timeouts initial** *seconds*	Specify the number of seconds the system waits for the caller to input the first digit of the dialed digits. Valid entries for this command are from 0 to 120.
Step 7	**timeouts interdigit** *seconds*	Specify the number of seconds the system waits (after the caller has input the initial digit) for the caller to input a subsequent digit. Valid entries for this command are from 0 to 120.
Step 8	**timing digit** *milliseconds*	If the voice-port dial type is DTMF, configure the DTMF digit signal duration. The range of the DTMF digit signal duration is from 50 to 100. The default is 100.
Step 9	**timing inter-digit** *milliseconds*	If the voice-port dial type is DTMF, configure the DTMF inter-digit signal duration. The range of the DTMF inter-digit signal duration is from 50 to 500. The default is 100.
Step 10	**timing pulse-digit** *milliseconds*	(FXO ports only.) If the voice-port dial type is pulse, configure the pulse-digit signal duration. The range of the pulse-digit signal duration is from 10 to 20. The default is 20.
Step 11	**timing pulse-inter-digit** *milliseconds*	(FXO ports only.) If the voice-port dial type is pulse, configure the pulse inter-digit signal duration. The range of the pulse-inter-digit signal duration is from 100 to 1000. The default is 500.

Note After you change any voice-port command, it is a good idea to cycle the port by using the **shutdown** and **no shutdown** commands.

Configure E&M Voice Ports

Unlike FXO and FXS voice ports, the default E&M voice-port parameters most likely are not sufficient to enable voice data transmission over your IP network. E&M voice-port values must match those specified by the particular PBX device to which the voice port is connected.

To configure an E&M voice port, perform the following steps. Items included in Step 1 and Step 2 are required; items included in Step 3 are optional.

Step 1 Identify the voice port and enter the voice-port configuration mode by using the **voice-port** command.

Step 2 For each of the following required parameters, select the appropriate parameter value by using the commands indicated:

- Dial type using the **dial-type** command
- Signal type using the **signal** command
- Call progress tone using the **cptone** command
- Operation using the **operation** command
- Type using the **type** command
- Impedance using the **impedance** command

Step 3 Select one or more of the following optional parameters by using the indicated commands:

- PLAR connection mode using the **connection plar** command
- Music threshold using the **music-threshold** command
- Description using the **description** command
- Comfort tone (if VAD is activated) using the **comfort-noise** command

To configure E&M voice ports, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	**configure terminal**	Enter the global configuration mode.
Step 2	**voice-port** *slot-number/subunit-number/port*	Identify the voice port you want to configure and enter the voice-port configuration mode.
Step 3	**dial-type** {**dtmf** | **pulse**}	Select the appropriate dial type for out-dialing.
Step 4	**signal** {**wink-start** | **immediate** | **delay-dial**}	Select the appropriate signal type for this interface.
Step 5	**cptone** {**australia** | **brazil** | **china** | **finland** | **france** | **germany** | **japan** | **northamerica** | **unitedkingdom**}	Select the appropriate voice call progress tone for this interface.

	Command	Purpose
Step 6	**connection plar** *string*	(Optional.) Specify the Private Line Auto Ringdown (PLAR) connection if this voice port is used for a PLAR connection. The *string* value specifies the destination telephone number.
Step 7	**music-threshold** *number*	(Optional.) Specify the threshold (in decibels) for on-hold music. Valid entries are from –70 dB to –30 dB.
Step 8	**description** *string*	(Optional.) Attach descriptive text about this voice-port connection.
Step 9	**comfort-noise**	(Optional.) Specify that background noise will be generated.

Validation Tips

You can check the validity of your voice-port configuration by performing the following tasks:

- Pick up the handset of an attached telephony device and check for a dial tone.
- If you have a dial tone, check for DTMF detection. If the dial tone stops when you dial a digit, then the voice port is most likely configured properly.
- Use the **show voice-port** command to verify that the data configured is correct.

Troubleshooting Tips

If you are having trouble connecting a call and you suspect the problem is associated with voice-port configuration, you can try to resolve the problem by performing the following tasks:

- Ping the associated IP address to confirm connectivity. If you cannot successfully ping your destination, refer to the *Network Protocols Configuration Guide, Part 1*.
- Use the **show voice-port** command to make sure that the port is enabled. If the port is offline, use the **no shutdown** command.
- If you have configured E&M interfaces, make sure that the values pertaining to your specific PBX setup, such as timing or type, are correct.
- Check to see if the voice network module has been correctly installed. For more information, refer to the installation document that came with your voice network module.

Fine-Tune E&M Voice Ports

Depending on the specifics of your particular network, you may need to adjust voice parameters involving timing, input gain, and output attenuation for E&M voice ports. Collectively, these commands are referred to as voice-port tuning commands.

Note In most cases, the default values for voice-port tuning commands are sufficient.

To configure voice-port tuning for E&M voice ports, perform the following steps:

Step 1 Identify the voice port and enter the voice-port configuration mode by using the **voice-port** command.

Step 2 For each of the following parameters, select the appropriate value using the commands indicated:

- Input gain using the **input gain** command
- Output attenuation using the **output attenuation** command
- Echo cancel coverage using the **echo-cancel enable** and **echo-cancel coverage** commands
- Nonlinear processing using the **non-linear** command
- Initial digit timeouts using the **timeouts initial** command
- Interdigit timeouts using the **timeouts interdigit** command
- Timing other than timeouts using the **timing clear-wait**, **timing delay-duration**, **timing delay-start**, **timing dial-pulse min-delay**, **timing digit**, **timing inter-digit**, **timing pulse**, **timing pulse-inter-digit**, **timing wink-duration**, and **timing wink-wait** commands

To fine-tune E&M voice ports, use the following commands beginning in privileged EXEC mode:

	Command	Purpose
Step 1	**configure terminal**	Enter the global configuration mode.
Step 2	**voice-port** *slot-number/subunit-number/port*	Identify the voice port you want to configure and enter the voice-port configuration mode.
Step 3	**echo-cancel enable**	Enable echo cancellation of voice that is sent out the interface and received back on the same interface.
Step 4	**echo-cancel coverage** *value*	Adjust the size (in milliseconds) of the echo cancel. Acceptable values are 16, 24, and 32.
Step 5	**non-linear**	Enable nonlinear processing, which shuts off any signal if no near-end speech is detected. (Nonlinear processing is used with echo cancellation.)
Step 6	**timeouts initial** *seconds*	Specify the number of seconds the system waits for the caller to input the first digit of the dialed digits. Valid entries for this command are from 0 to 120.

	Command	Purpose
Step 7	**timeouts interdigit** *seconds*	Specify the number of seconds the system waits (after the caller has input the initial digit) for the caller to input a subsequent digit. Valid entries for this command are from 0 to 120.
Step 8	**timing clear-wait** *milliseconds* **timing delay-duration** *milliseconds* **timing delay-start** *milliseconds* **timing dial-pulse min-delay** *milliseconds* **timing digit** *milliseconds* **timing inter-digit** *milliseconds* **timing pulse** *pulse-per-second* **timing pulse-inter-digit** *milliseconds* **timing wink-duration** *milliseconds* **timing wink-wait** *milliseconds*	Specify timing parameters. Valid entries for **clear-wait** are from 200 to 2000 milliseconds (ms). Valid entries for **delay-duration** are from 100 to 5000 ms. Valid entries for **delay-start** are from 20 to 2000 ms. Valid entries for **dial-pulse min-delay** are from 0 to 5000 ms. Valid entries for **digit** are from 50 to 100 ms. Valid entries for **inter-digit** are from 50 to 500 ms. Valid entries for **pulse** are from 10 to 20 pulse-per-second. Valid entries for **pulse-inter-digit** are 100 to 1000 ms. Valid entries for **wink-duration** are from 100 to 400 ms. Valid entries for **wink-wait** are from 100 to 5000 ms.

Note After you change any voice-port command, it is a good idea to cycle the port by using the **shutdown** and **no shutdown** commands.

Optimize Dial Peer and Network Interface Configurations

Depending on how you have configured your network interfaces, you might need to configure additional VoIP dial-peer parameters. This section describes the following topics:

- Configure IP Precedence for Dial Peers, page 4-29
- Configure RSVP for Dial Peers, page 4-30
- Configure Codec and VAD for Dial Peers, page 4-31

Configure IP Precedence for Dial Peers

If you want to give real-time voice traffic a higher priority than other network traffic, you can weight the voice data traffic associated with a particular VoIP dial peer by using IP precedence. IP precedence scales better than RSVP but provides no admission control.

To give real-time voice traffic precedence over other IP network traffic, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	**dial-peer voice** *number* **voip**	Enter the dial-peer configuration mode to configure a VoIP peer.
Step 2	**ip precedence** *number*	Select a precedence level for the voice traffic associated with that dial peer.

In IP precedence, the numbers 1 through 5 identify classes for IP flows; the numbers 6 and 7 are used for network and backbone routing and updates.

For example, to ensure that voice traffic associated with VoIP dial peer 103 is given a higher priority than other IP network traffic, enter the following:

```
dial-peer voice 103 voip
 ip precedence 5
```

In this example, when an IP call leg is associated with VoIP dial peer 103, all packets transmitted to the IP network through this dial peer have their precedence bits set to 5. If the networks receiving these packets have been configured to recognize precedence bits, the packets are given priority over packets with a lower configured precedence value.

Configure RSVP for Dial Peers

If you have configured your WAN or LAN interfaces for RSVP, you must configure the QoS for any associated VoIP peers. To configure quality of service for a selected VoIP peer, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	**dial-peer voice** *number* **voip**	Enter the dial-peer configuration mode to configure a VoIP peer.
Step 2	**req-qos** [**best-effort** | **controlled-load** | **guaranteed-delay**]	Specify the desired quality of service to be used.

Note We suggest that you select **controlled-load** for the requested quality of service.

For example, to specify guaranteed delay QoS for VoIP dial peer 108, enter the following:

```
dial-peer voice 108 voip
 destination-pattern +1408528
 req-qos controlled-load
 session target ipv4:10.0.0.8
```

In this example, every time a connection is made through VoIP dial peer 108, an RSVP reservation request is made between the local router, all intermediate routers in the path, and the final destination router.

To generate an SNMP trap message if the reserved QoS is less than the configured value for a selected VoIP peer, use the following commands, beginning from the global configuration mode:

	Command	Purpose
Step 1	**dial-peer voice** *number* **voip**	Enter the dial-peer configuration mode to configure a VoIP peer.
Step 2	**acc-qos** [**best-effort** | **controlled-load** | **guaranteed-delay**]	Specify the QoS value below which an SNMP trap will be generated.

Note RSVP reservations are only one-way. If you configure RSVP, the VoIP dial peers on both ends of the connection must be configured for RSVP.

Configure Codec and VAD for Dial Peers

Coder-decoder (codec) and voice activity detection (VAD) for a dial peer determine how much bandwidth the voice session uses. Typically, codec is used to transform analog signals into a digital bit stream and digital signals back into analog signals—in this case, it specifies the voice coder rate of speech for a dial peer. VAD is used to disable the transmission of silence packets.

Configure Codec for a VoIP Dial Peer

To specify a voice coder rate for a selected VoIP peer, use the following commands, initially beginning in global configuration mode:

	Command	Purpose
Step 1	**dial-peer voice** *number* **voip**	Enter the dial-peer configuration mode to configure a VoIP peer.
Step 2	**codec** [**g711alaw** | **g711ulaw** | **g729r8**]	Specify the desired voice coder rate of speech.

The default for the **codec** command is **g729r8**; normally the default configuration for this command is the most desirable. If, however, you are operating on a high-bandwidth network and voice quality is of the highest importance, you should configure the **codec** command for **g711alaw** or **g711ulaw**. Using these values results in better voice quality, but it also requires higher bandwidth requirements for voice.

For example, to specify a codec rate of G.711a-law for VoIP dial peer 108, enter the following:

```
dial-peer voice 108 voip
 destination-pattern +1408528
 codec g711alaw
 session target ipv4:10.0.0.8
```

Configure VAD for a VoIP Dial Peer

To disable the transmission of silence packets for a selected VoIP peer, use the following commands, beginning in global configuration mode:

	Command	Purpose
Step 1	**dial-peer voice** *number* **voip**	Enter the dial-peer configuration mode to configure a VoIP peer.
Step 2	**vad**	Disable the transmission of silence packets (enabling VAD).

The default for the **vad** command is enabled; normally the default configuration for this command is the most desirable. If you are operating on a high-bandwidth network and voice quality is of the highest importance, you should disable vad. Using this value will result in better voice quality, but it will also require higher bandwidth requirements for voice.

For example, to enable VAD for VoIP dial peer 108, enter the following:

```
dial-peer voice 108 voip
destination-pattern +1408528
vad
session target ipv4:10.0.0.8
```

Configuring Voice over Frame Relay

You need to take certain factors into consideration when configuring VoIP for it to run smoothly over Frame Relay. A public Frame Relay cloud provides no guarantees for QoS. For real-time traffic to be transmitted in a timely manner, the data rate must not exceed the committed information rate (CIR), or there is the possibility that packets will be dropped. In addition, Frame Relay traffic shaping and RSVP are mutually exclusive. This is particularly important to remember if multiple data-link connection identifiers (DLCIs) are carried on a single interface.

For Frame Relay links with slow output rates (less than or equal to 64 kbps), where data and voice are being transmitted over the same PVC, we recommend the following solutions:

- Separate DLCIs for voice and data—By providing a separate subinterface for voice and data, you can use the appropriate QoS tool per line. For example, each DLCI would use 32 kbps of a 64 kbps line.
  - Apply adaptive traffic shaping to both DLCIs.
  - Use RSVP or IP precedence to prioritize voice traffic.
  - Use compressed RTP to minimize voice packet size.
  - Use weighted fair queuing to manage voice traffic.
- Lower MTU size—Voice packets are generally small. By lowering the maximum transmission unit (MTU) size (for example, to 300 bytes), large data packets can be broken up into smaller data packets that can more easily be interwoven with voice packets.

Note Lowering the MTU size affects data throughput speed.

- CIR equal to line rate—Make sure that the data rate does not exceed the CIR. This is accomplished through generic traffic shaping.
  - Use RSVP or IP precedence to prioritize voice traffic.
  - Use compressed RTP to minimize voice packet header size.
- Traffic shaping—Use adaptive traffic shaping to throttle back the output rate based on the backward explicit congestion notification (BECN). If the feedback from the switch is ignored, packets (both data and voice) might be discarded. Because the Frame Relay switch does not distinguish between voice and data packets, voice packets could be discarded, which would result in a deterioration of voice quality.
  - Use RSVP, compressed RTP, reduced MTU size, and adaptive traffic shaping based on BECN to hold the data rate to the CIR.
  - Use generic traffic shaping to obtain a low interpacket wait time. For example, set Bc to 4000 to obtain an interpacket wait of 125 ms.

Voice over Frame Relay Configuration Example

For Frame Relay, it is customary to configure a main interface and several subinterfaces, one subinterface per PVC. The following example configures a Frame Relay main interface and a subinterface so that voice and data traffic can be successfully transported:

```
interface serial 1/1
MTU 300
no ip address
encapsulation frame-relay
no ip route-cache
no ip mroute-cache
fair-queue 64 256 1000
frame-relay ip rtp header-compression

interface serial 1/1.1
MTU 300
ip address 10.0.0.5 255.0.0.0
ip rsvp bandwidth 48 48
no ip route-cache
no ip mroute-cache
bandwidth 64
traffic-shape rate 32000 4000 4000
frame-relay interface-dlci 16
frame-relay ip rtp header-compression
```

In this configuration example, the main interface has been configured as follows:

- MTU size is 300 bytes.
- No IP address is associated with this serial interface. The IP address must be assigned for the subinterface.
- Encapsulation method is Frame Relay.
- Fair queuing is enabled.
- IP RTP header compression is enabled

The subinterface has been configured as follows:

- MTU size is inherited from the main interface.
- IP address for the subinterface is specified.
- Bandwidth is set to 64 kbps.
- RSVP is enabled to use the default value, which is 75 percent of the configured bandwidth.
- Generic traffic shaping is enabled with 32 kbps CIR where Bc = 4000 bits and Be = 4000 bits.
- Frame Relay DLCI number is specified.
- IP RTP header compression is enabled.

Note When traffic bursts over the CIR, output rate is held at the speed configured for the CIR (for example, traffic will not go beyond 32 kbps if the CIR is set to 32 kbps).

For more information about Frame Relay, refer to the Cisco IOS Release 12.0 *Wide-Area Networking Configuration Guide*.

Checking the Configuration

After configuring the new interface, use the **show** commands to display the status of the new interface or all interfaces, and use the **ping** command to check connectivity. This section includes the following subsections:

- Using show Commands to Verify the New Interface Status, page 4-34
- Using loopback Commands, page 4-42

Using show Commands to Verify the New Interface Status

Table 4-4 demonstrates how you can use the **show** commands to verify that new interfaces are configured and operating correctly and that the PA-VXA, PA-VXB, or PA-VXC port adapter appears in them correctly. Sample displays of the output of selected **show** commands appear in the sections that follow. For complete command descriptions and examples, refer to the publications listed in the "Related Documentation" section on page viii.

Troubleshooting Tips

If information about the port adapter is not indicated in show command output, it is probably because the card type has not been specified.

Because the port adapter can be configured for E1 or T1 connectivity, you must specify the card type as E1 or T1, as described in "Performing a Basic Configuration" section on page 4-4. There is no default card type. The port adapter is not functional until the card type is set.

Note The outputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Table 4-4 Using show Commands

Command	Function	Example
show version or **show hardware**	Displays system hardware configuration, the number of each interface type installed, Cisco IOS software version, names and sources of configuration files, and boot images	`Router# show version`
show controllers	Displays all the current interface processors and their interfaces	`Router# show controllers`
show diag *slot*	Displays types of port adapters installed in your system and information about a specific port adapter slot, interface processor slot, or chassis slot	`Router# show diag 2`

Table 4-4 Using show Commands (continued)

Command	Function	Example
show interfaces *type port-adapter-slot-number/ interface-port-number*	Displays status information about a specific type of interface (for example, DSPfarm) in Cisco 7200 series routers, Cisco 7200 VXR routers, Cisco 7401ASR routers, or Cisco 7500 series routers	`Router# show interfaces DSPfarm 1/0`
show protocols	Displays protocols configured for the entire system and for specific interfaces	`Router# show protocols`
show running-config	Displays the running configuration file	`Router# show running-config`
show startup-config	Displays the configuration stored in NVRAM	`Router# show startup-config`

If an interface is shut down and you configured it as up, or if the displays indicate that the hardware is not functioning properly, ensure that the interface is properly connected and terminated. If you still have problems bringing up the interface, contact a service representative for assistance. This section includes the following subsections and offers some platform-specific output examples:

- Using the show version or show hardware Commands, page 4-35
- Using the show diag Command, page 4-37
- Using the show interfaces Command, page 4-40

Choose the subsection appropriate for your system. Proceed to the "Using loopback Commands" section on page 4-42 when you have finished using the **show** commands.

Using the show version or show hardware Commands

Use the **show version** (or **show hardware**) command to display the configuration of the system hardware, the number of each interface type installed, the Cisco IOS software version, the names and sources of configuration files, and the boot images.

Note The outputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Cisco 7200 Series and Cisco 7200 VXR Routers

Following is an example of the **show version** command from a Cisco 7206VXR router:

```
Router# show version

Cisco Internetwork Operating System Software
IOS (tm) 7200 Software (C7200-IS-M), Version 12.0(5)XE3,
TAC:Home:SW:IOS:Specials for info
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Sat 20-Nov-99 10:51 by en
Image text-base: 0x60008900, data-base: 0x6139A000
```

```
ROM: System Bootstrap, Version 12.0(19990210:195103) [12.0XE 105],
BOOTFLASH: 7200 Software (C7200-IS-M), Version 12.0(5)XE3,

Router uptime is 4 days, 21 hours, 8 minutes
System returned to ROM by reload
System image file is "slot0:c7200-is-mz.120-5.XE3"

cisco 7206VXR (NPE300) processor with 57344K/40960K bytes of memory.
R7000 CPU at 262Mhz, Implementation 39, Rev 1.0, 256KB L2, 2048KB L3 Cache
6 slot VXR midplane, Version 2.0

Last reset from power-on
Bridging software.
X.25 software, Version 3.0.0.
Primary Rate ISDN software, Version 1.1.
1 FastEthernet/IEEE 802.3 interface(s)
1 ATM network interface(s)
10 Channelized T1/PRI port(s)
5 Voice resource(s)
125K bytes of non-volatile configuration memory.

20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash PCMCIA card at slot 1 (Sector size 128K).
4096K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x0
```

Cisco 7401ASR Routers

Following is an example of the **show version** command from a Cisco 7401ASR router with the PA-VXB-2TE1+ port adapter shown as a voice resource:

```
Router# show version

Cisco Internetwork Operating System Software
IOS (tm) 7401ASR Software (C7401ASR-IS-M), Version 12.0(5)XE3,
TAC:Home:SW:IOS:Specials for info
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Sat 20-Nov-99 10:51 by en
Image text-base: 0x60008900, data-base: 0x6139A000

ROM: System Bootstrap, Version 12.0(19990210:195103) [12.0XE 105],
BOOTFLASH: 7401ASR Software (C7401ASR-IS-M), Version 12.0(5)XE3,

Router uptime is 4 days, 21 hours, 8 minutes
System returned to ROM by reload
System image file is "slot1:c7401ASR-is-mz.120-5.XE3"

Last reset from power-on
Bridging software.
X.25 software, Version 3.0.0.
Primary Rate ISDN software, Version 1.1.
1 FastEthernet/IEEE 802.3 interface(s)
1 ATM network interface(s)
10 Channelized T1/PRI port(s)
5 Voice resource(s)
125K bytes of non-volatile configuration memory.

20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
8192K bytes of Flash PCMCIA card at slot 1 (Sector size 128K).
4096K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x0
```

Cisco 7500 Series Routers

Following is an example of the **show version** command from a Cisco 7500 series router:

```
Router# show version

Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-ISV-M), Version 12.1(20000613:174753) []
Copyright (c) 1986-2000 by cisco Systems, Inc.
Compiled Wed 28-Jun-00 18:59 by ang
Image text-base: 0x60010968, data-base: 0x613FC000

ROM: System Bootstrap, Version 5.3(16645) [ang 571],RELEASE SOFTWARE
BOOTFLASH: GS Software (RSP-BOOT-M), Version 11.1(5654)XX [mwu 105]

cheng10 uptime is 14 hours, 6 minutes
System returned to ROM by power-on
System image file is "tftp://223.255.254.254/muck/ang/rsp-isv-mz.1213t"

cisco RSP2 (R4600) processor with 81920K/2072K bytes of memory.
R4600 CPU at 100Mhz, Implementation 32, Rev 2.0
Last reset from power-on
G.703/E1 software, Version 1.0.
G.703/JT2 software, Version 1.0.
X.25 software, Version 3.0.0.
Bridging software.
Primary Rate ISDN software, Version 1.1.
Chassis Interface.
1 VIP2 controller (4 T1)(1 ATM)(2 Voice resources).
2 VIP2 R5K controllers (8 Ethernet)(4 T1)(1 ATM)(2 Voice resources).
8 Ethernet/IEEE 802.3 interface(s)
24 Serial network interface(s)
2 ATM network interface(s)
125K bytes of non-volatile configuration memory.

20480K bytes of Flash PCMCIA card at slot 0 (Sector size 128K).
20480K bytes of Flash PCMCIA card at slot 1 (Sector size 128K).
8192K bytes of Flash internal SIMM (Sector size 256K).
Configuration register is 0x100
```

Using the show diag Command

Display the types of port adapters installed in your system (and specific information about each) using the **show diag** *slot* command, where *slot* is the *port adapter slot* in Cisco 7200 series routers, Cisco 7200 VXR routers, and Cisco 7401ASR routers, and the *interface processor slot* in a Cisco 7500 series router.

Note The outputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Cisco 7200 Series and Cisco 7200 VXR Routers

Following is an example of the **show diag** *slot* command that shows a PA-VXC-2TE1 in port adapter slot 1 of a Cisco 7200 series router:

```
Router# show diag 1
Slot 1:
        VXC-2T1/E1 Port adapter, 2 ports
        Port adapter is analyzed
```

```
        Port adapter insertion time 4d21h ago
        EEPROM contents at hardware discovery:
        Hardware revision 1.0           Board revision K0
        Serial number     14320311      Part number    73-3861-03
        Test history      0x0           RMA number     00-00-00
        EEPROM format version 1
        EEPROM contents (hex):
          0x20: 01 CF 01 00 00 DA 82 B7 49 0F 15 03 00 00 00 00
          0x30: A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Note Port adapters used with Cisco 7200 VXR routers require the correct base hardware revision in order to function. Use the **show diag** command to display the hardware revision.

Cisco 7401ASR Routers

Following is an example of the **show diag** *slot* command that shows a PA-VXB-2TE1+ port adapter in port adapter slot 1 of a Cisco 7401ASR router:

```
Router# show diag 1
Slot 1:
        VXB-2T1/E1+ Port adapter, 2 ports
        Port adapter is analyzed
        Port adapter insertion time 4d21h ago
        EEPROM contents at hardware discovery:
        Hardware revision 1.0           Board revision K0
        Serial number     14320311      Part number    73-3861-03
        Test history      0x0           RMA number     00-00-00
        EEPROM format version 1
        EEPROM contents (hex):
          0x20: 01 CF 01 00 00 DA 82 B7 49 0F 15 03 00 00 00 00
          0x30: A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Cisco 7500 Series Routers

Following is an example of the **show diag** *slot* command that shows a PA-VXC-2TE1+ in interface processor slot 1 of a Cisco 7500 series router:

```
Router# show diag 1
Slot 1:
        Physical slot 1, ~physical slot 0xE, logical slot 1, CBus 0
        Microcode Status 0x4
        Master Enable, LED, WCS Loaded
        Board is analyzed
        Pending I/O Status: None
        EEPROM format version 1
        VIP2 R5K controller, HW rev 2.02, board revision A0
        Serial number: 10945159  Part number: 73-2167-05
        Test history: 0x00        RMA number: 00-00-00
        Flags: cisco 7000 board; 7500 compatible

        EEPROM contents (hex):
          0x20: 01 1E 02 02 00 A7 02 87 49 08 77 05 00 00 00 00
          0x30: 50 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00

        Slot database information:
        Flags: 0x4      Insertion time: 0x288C (3d19h ago)

        Controller Memory Size: 128 MBytes DRAM, 8192 KBytes SRAM

        PA Bay 0 Information:
```

```
VXC-2T1/E1+ Port adapter, 2 ports
EEPROM format version 4
HW rev 0.02, Board revision A0
Serial number: MIC04251WXZ  Part number: 73-5340-03
```

```
PA Bay 1 Information:
        VXC-2T1/E1 Port adapter, 2 ports
        EEPROM format version 1
        HW rev 0.01, Board revision A0
        Serial number: 15234366  Part number: 73-3861-03
```

Using the show interfaces Command

The **show interfaces** command displays status information (including the physical slot and interface address) for the interfaces you specify. All of the examples that follow specify DSPfarm interfaces.

For complete descriptions of interface subcommands and the configuration options available for router interfaces, refer to the publications listed in the "Related Documentation" section on page viii.

Note The outputs that appear in this document may not match the output you receive when running these commands. The outputs in this document are examples only.

Cisco 7200 Series and Cisco 7200 VXR Routers

Following is an example of the **show interfaces DSPfarm** command for Cisco 7200 series routers. In this example, the PA-VXC-2TE1 port adapter is located in port adapter slot 1.

```
Router# show interfaces DSPfarm 1/0
DSPfarm1/0 is up, line protocol is up
  Hardware is VXC-2T1/E1
  MTU 256 bytes, BW 12000 Kbit, DLY 0 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  C549 DSP Firmware Version: MajorRelease.MinorRelease (BuildNumber)
     DSP Boot Loader: 255.255 (255)
     DSP Application: 3.2 (10)
     Medium Complexity Application: 3.2 (12)
     High Complexity Application: 3.2 (12)
  Total DSPs 29, DSP0-DSP29, Jukebox DSP id 30
  Down DSPs: none
  Restarted DSPs due to background keep-alive failures DSPid(number of restart)e
  Total sig channels 116 used 48, total voice channels 116 used 0
     0 active calls, 24 max active calls, 163554 total calls
     CHT: 489702 total sec, 2 avg sec, 69 max sec
     230728928 rx packets, 0 rx drops, 239246561 tx packets, 0 tx frags
     0 curr_dsp_tx_queued, 12 max_dsp_tx_queued
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 4d20h
  Queueing strategy: fifo
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     228929291 packets input, 574310582 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     239246561 packets output, 626450944 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Cisco 7401ASR Routers

Following is an example of the **show interfaces DSPfarm** command for Cisco 7401ASR routers. In this example, the PA-VXB-2TE1+ port adapter is located in port adapter slot 1.

```
Router# show interfaces DSPfarm 1/0
DSPfarm1/0 is up, line protocol is up
  Hardware is VXB-2T1/E1+
  MTU 256 bytes, BW 12000 Kbit, DLY 0 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  C549 DSP Firmware Version: MajorRelease.MinorRelease (BuildNumber)
     DSP Boot Loader: 255.255 (255)
     DSP Application: 3.2 (10)
     Medium Complexity Application: 3.2 (12)
     High Complexity Application: 3.2 (12)
  Total DSPs 29, DSP0-DSP29, Jukebox DSP id 30
  Down DSPs: none
  Restarted DSPs due to background keep-alive failures DSPid(number of restart)e

  Total sig channels 116 used 48, total voice channels 116 used 0
     0 active calls, 24 max active calls, 163554 total calls
     CHT: 489702 total sec, 2 avg sec, 69 max sec
     230728928 rx packets, 0 rx drops, 239246561 tx packets, 0 tx frags
     0 curr_dsp_tx_queued, 12 max_dsp_tx_queued
  Last input never, output never, output hang never
  Last clearing of "show interface" counters 4d20h
  Queueing strategy: fifo
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     228929291 packets input, 574310582 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     239246561 packets output, 626450944 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Cisco 7500 Series Routers

Following is an example of the **show interfaces DSPfarm** command for Cisco 7500 series routers. In this example, the PA-VXC-2TE1+ port adapter is located in port adapter slot 0 on a VIP motherboard installed in interface processor slot 1.

```
Router# show interfaces DSPfarm 1/0/0
DSPfarm1/0/0 is up, line protocol is up
  Hardware is VXC-2T1/E1-PLUS
  MTU 256 bytes, BW 12000 Kbit, DLY 0 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation VOICE, loopback not set
  C549 DSP Firmware Version: MajorRelease.MinorRelease (BuildNumber)
     DSP Boot Loader: 3.4 (1)
     DSP Application: 3.2 (10)
     Medium Complexity Application: 3.4 (25)
     High Complexity Application: 3.4 (25)
  Total DSPs 30, DSP0-DSP29, Jukebox DSP id 30
  Down DSPs: none
  DSPs restarted: none
  Total sig channels 120 used 47, total voice channels 120 used 24
     24 active calls, 47 max active calls, 1196 total calls
     CHT: 3566 total sec, 2 avg sec, 33 max sec
     0 curr_dsp_tx_queued, 22 max_dsp_tx_queued
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 85000 bits/sec, 65 packets/sec
  5 minute output rate 75000 bits/sec, 65 packets/sec
```

```
5651409 packets input, 683571491 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
5698903 packets output, 586641447 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
```

Proceed to "Using loopback Commands" section on page 4-42 to check network connectivity of the PA-VXB or PA-VXC and the switch or router.

Using loopback Commands

With the loopback test, you can detect and isolate equipment malfunctions by testing the connection between the PA-VXA, PA-VXB, or PA-VXC interface and a remote device such as a modem or a CSU/DSU. The **loopback** subcommand places an interface in loopback mode, which enables test packets that are generated from the **ping** command to loop through a remote device or compact serial cable. If the packets complete the loop, the connection is good. If not, you can isolate a fault to the remote device or compact serial cable in the path of the loopback test.

Depending on the mode of the port, issuing the **loopback** command checks the following paths:

- When no compact serial cable is attached to the PA-VXB or PA-VXC interface port, or if a DCE cable is attached to a port that is configured as line protocol up, the **loopback** command tests the path between the network processing engine and the interface port only (without leaving the network processing engine and port adapter).
- When a DTE cable is attached to the port, the **loopback** command tests the path between the network processing engine and the near (network processing engine) side of the DSU or modem to test the PA-VXB or PA-VXC interface and compact serial cable.

T1 Loopback Examples

Specify loopback for a T1 controller and T1 channel using the **loopback** command. There are three main loopback modes: diagnostic, local (line and payload), and remote (IBOC and ESF). Specify the loopback format using the **loopback** [**diagnostic | local | remote**] command.

Note To shut down the T1 controller, use the **shutdown** command at the controller prompt.

Examples of specific loopback modes for the T1 controller follow:

- The syntax of the **loopback diagnostic** command is as follows:

  loopback [**diagnostic**]

  Set the first T1 into diagnostic loopback.

  Cisco 7200 Series Routers:

  ```
  Router# config t
  Enter configuration commands, one per line. End with CNTL/Z.
  Router(config)# controller t1 2/0
  Router(config-controller)# loopback diagnostic
  ```

Cisco 7401ASR Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller t1 1/0
Router(config-controller)# loopback diagnostic
```

Cisco 7500 Series Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller t1 2/2/0
Router(config-controller)# loopback diagnostic
```

In the preceding examples, the **loopback diagnostic** command loops the outgoing transmit signal back to the receive signal and sends an alarm indication signal (AIS) to the network.

- The syntax of the **loopback local** command is as follows:

```
loopback [local {payload | line}]
```

Set the first T1 into local loopback.

Cisco 7200 Series Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller t1 2/0
Router(config-controller)# loopback local payload
```

Cisco 7401ASR Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller t1 1/0
Router(config-controller)# loopback local payload
```

Cisco 7500 Series Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller t1 2/2/0
Router(config-controller)# loopback local payload
```

In the preceding examples, the **loopback local payload** command loops the incoming signal back to the line.

- The syntax of the **loopback remote** command follows:

```
loopback [remote {esf line | iboc | esf payload}]
```

Set the first T1 into remote line inband loopback.

Cisco 7200 Series Routers:

```
Router# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# controller t1 2/0
Router(config-controller)# loop remote esf line
```

Cisco 7401ASR Routers:

```
Router# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# controller t1 1/0
Router(config-controller)# loop remote esf line
```

Cisco 7500 Series Routers:

```
Router# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# controller t1 2/2/0
Router(config-controller)# loop remote esf line
```

This command causes the far end to loop its receive signal back to transmit.

E1 Loopback Examples

Specify loopback for an E1 controller using the **loopback** command. There are two main loopback modes: diagnostic and local (line and payload). Specify the loopback format using the **loopback** [**diagnostic** | **local**] command.

Note To shut down the E1 controller, use the **shutdown** command at the controller prompt.

Examples of specific loopback modes for the E1 controller follow:

- The syntax of the **loopback diagnostic** command is as follows:

  ```
  loopback [diagnostic]
  ```

  Set the first E1 into diagnostic loopback.

  Cisco 7200 Series Routers:

  ```
  Router# config t
  Enter configuration commands, one per line. End with CNTL/Z.
  Router(config)# controller E1 2/0
  Router(config-controller)# loopback diagnostic
  ```

  Cisco 7401ASR Routers:

  ```
  Router# config t
  Enter configuration commands, one per line. End with CNTL/Z.
  Router(config)# controller E1 1/0
  Router(config-controller)# loopback diagnostic
  ```

  Cisco 7500 Series Routers:

  ```
  Router# config t
  Enter configuration commands, one per line. End with CNTL/Z.
  Router(config)# controller E1 2/2/0
  Router(config-controller)# loopback diagnostic
  ```

  In the preceding examples, the **loopback diagnostic** command loops the outgoing transmit signal back to the receive signal and sends an AIS out to the network.

- The syntax of the **loopback local** command is as follows:

  ```
  loopback [local {payload | line}]
  ```

  Set the first E1 into local loopback.

  Cisco 7200 Series Routers:

  ```
  Router# config t
  Enter configuration commands, one per line. End with CNTL/Z.
  Router(config)# controller E1 2/0
  Router(config-controller)# loopback local payload
  ```

Cisco 7401ASR Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller E1 1/0
Router(config-controller)# loopback local payload
```

Cisco 7500 Series Routers:

```
Router# config t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# controller E1 2/2/0
Router(config-controller)# loopback local payload
```

In the preceding examples, the **loopback local** command loops the incoming signal back to the line.