Nways Multiprotocol Routing Services

Software User's Guide Version 2.1

SC30-3681-05

Nways Multiprotocol Routing Services

Software User's Guide Version 2.1

SC30-3681-05

Note

Before using this document, read the general information under “Notices” on page iii.

Sixth Edition (October 1997)

This edition applies to Version 2.1 of the IBM Nways Multiprotocol Routing Services and to all subsequent releases and modifications until otherwise indicated in new editions or technical newsletters.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

IBM welcomes your comments. A form for readers' comments is provided at the back of this publication. If the form has been removed, you may address your comments to:

Department CGF
Design & Information Development
IBM Corporation
P.O. Box 12195
RESEARCH TRIANGLE PARK NC 27709
USA

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

Advanced Peer-to-Peer Networking	IBM	PS/2
AIX	Micro Channel	RS/6000
AIXwindows	NetView	System/370
APPN	Nways	VTAM
BookManager		

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Microsoft, Windows, Windows NT, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

Other company, product, and service names may be trademarks or service marks of others.

Contents

Figures

Tables

Preface

Preface

This manual contains the information that you will need to use the router user interface for configuration and operation of the Multiprotocol Routing Services base code installed on your Nways device. With the help of this manual, you should be able to perform the following processes and operations:

- Configure, monitor, and use the Multiprotocol Routing Services base code.
- Configure, monitor, and use the interfaces and Link Layer software supported by your Nways device.

This manual contains the information you will need to configure bridging and routing functions on an Nways device. The manual describes all of the features and functions that are in the software. A specific Nways device might not support all of the features and functions described. If a feature or function is device-specific, a notice in the relevant chapter or section indicates that restriction.

This manual supports the IBM 2210 and refers to this product as either “the routers” or “the device”. The examples in the manual represent the configuration of an IBM 2210 but the actual output you see may vary. Use the examples as a guideline to what you might see while configuring your device.

Who Should Read This Manual

This manual is intended for persons who install and manage computer networks. Although experience with computer networking hardware and software is helpful, you do not need programming experience to use the protocol software.

To Get Additional Information: Changes may be made to the documentation after the books are printed. If additional information is available or if changes are required after the books have been printed, the changes will be in a file (named README) on diskette 1 of the configuration program diskettes. You can view the file with an ASCII text editor.

About the Software

IBM Nways Multiprotocol Routing Services is the software that supports the IBM 2210 (licensed program number 5801-ARR). This software has these components:

- The base code, which consists of:
  - The code that provides the routing, bridging, data link switching, and SNMP agent functions for the device.
  - The router user interface, which allows you to configure, monitor, and use the Multiprotocol Routing Services base code installed on the device. The router user interface is accessed locally through an ASCII terminal or emulator attached to the service port, or remotely through a Telnet session or modem-attached device.

  The base code is installed at the factory on the 2210.

- The Configuration Program for IBM Nways Multiprotocol Routing Services (*Configuration Program*), a graphical user interface that allows you to configure the device from a stand-alone workstation. The Configuration Program includes error checking and online help information.

  The Configuration Program is not preloaded at the factory; it is shipped separately from the device as part of the software order.

  You can also FTP the Configuration Program for IBM Nways Multiprotocol Routing Services. See *Configuration Program User's Guide for Nways Multiprotocol Acess, Routing, and Switched Services,* GC30-3830, for the server address and directories.

Conventions Used in This Manual

The following conventions are used in this manual to show command syntax and program responses:

1. The abbreviated form of a command is shown in the following example:

   ```
   reload
   ```

   In this example, you can enter either the whole command (reload) or its abbreviation (rel).

2. Three periods following an option mean that you enter additional data (for example, a variable) after the option. For example:

   ```
   time host ...
   ```

   In this example, you enter the IP address of the host in place of the periods, as explained in the description of the command.

3. In information displayed in response to a command, defaults for an option are enclosed in brackets immediately following the option. For example:

   ```
   Media (UTP/STP) [UTP]
   ```

   In this example, the media defaults to UTP unless you specify STP.

4. Keyboard key combinations are indicated in text in the following ways:

   Ctrl **P**

IBM 2210 Nways Multiprotocol Router Publications

The following list shows the books that support the IBM 2210.

Operations and Network Management

SC30-3681 *Software User's Guide for Nways Multiprotocol Routing Services Version 2.1*

This book explains how to:

- Configure, monitor, and use the IBM Nways Multiprotocol Routing Services software shipped with the router.

- Use the Multiprotocol Routing Services command-line router user interface to configure and monitor the network interfaces and link-layer protocols shipped with the router.

SC30-3680 *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1*

SC30-3865 *Protocol Configuration and Monitoring Reference Volume 2 for Nways Multiprotocol Routing Services Version 2.1*

These books describe how to access and use the Multiprotocol Routing Services command-line router user interface to configure and monitor the routing protocol software shipped with the router.

They include information about each of the protocols that the devices support.

SC30-3682 *IBM Nways Event Logging System Messages Guide*

This book contains a listing of the error codes that can occur, along with descriptions and recommended actions to correct the errors.

Configuration

Online help The help panels for the Configuration Program assist the user in understanding the program functions, panels, configuration parameters, and navigation keys.

GC30-3830 *Configuration Program User's Guide for Nways Multiprotocol Acess, Routing, and Switched Services*

This book discusses how to use the Configuration Program.

GG24-4446 *IBM 2210 Nways Multiprotocol Router Description and Configuration Scenarios*

This book contains examples of how to configure protocols using IBM Nways Multiprotocol Routing Services.

Safety

SD21-0030 *Caution: Safety Information - Read This First*

This book provides translations of caution and danger notices applicable to the installation and maintenance of an IBM 2210.

The following list shows the books in the IBM 2210 Nways Multiprotocol Router library, arranged according to tasks.

Planning and Installation

GA27-4068 *IBM 2210 Nways Multiprotocol Router Installation and Initial Configuration Guide*

This book is shipped with the 2210, except models 1Sx and 1Ux. It explains how to prepare for installation, install the 2210, perform an initial configuration, and verify that the installation is successful.

This book provides translations of danger notices and other safety information.

GC30-3867 *IBM 2210 Models 1Sx and 1Ux Installation Guide*

This book is shipped with the 2210 Models 1Sx and 1Ux. It explains how to prepare for installation, install the 2210, perform an initial configuration and verify that the installation is successful.

This book provides translations of danger notices and other safety information.

Diagnostics and Maintenance

SY27-0345 *IBM 2210 Nways Multiprotocol Router Service and Maintenance Manual*

This book is shipped with the 2210. It provides instructions for diagnosing problems with and repairing the 2210.

Summary of Changes for the IBM 2210 Software Library

The changes for the IBM 2210 library consist of:

- **Information to support hardware**
  - New adapters:
    - 4-port S/T ISDN BRI adapter
    - 4-port U ISDN BRI adapter
    - All WAN interface types are now supported on the 4-port and 8-port WAN concentration adapters
- **Information to support software**
  - New Functions:
    - Dial-In Access to LAN Servers (DIALs)
    - Delete Interface command
    - Full WAN support for concentration adapters
    - Spare interface definition for dynamic reconfiguration
    - Ethernet Local MAC Address Administration
    - ISDN and WAN ports can now be active simultaneously on the 1Sx and 1Ux models. This function is useful for applications like WAN Restoral.
  - Enhanced Functions:
    - Bridging Enhancements
      - SR-TB duplicate MAC address support—allows a duplicate MAC address to exist in an SR domain and also offers a load-balancing feature.
    - DLSw Enhancements:
      - Circuit priority – allows you to set the circuit priority for a range of SAPs and MAC addresses.
      - MAC address list—allows you to build address lists for DLSw sessions to exchange as described in RFC 1795.
      - NetBIOS SessionAlive spoofing—allows you to disable TCP Keepalive messages for dial-on-demand circuits.
    - Serviceability Enhancements

- TFTP disable—to prevent unwanted updates or retrievals of device software from the network.
- ICMP redirect disable—prevents the device from transmitting a packet on the same interface on which it received the packet.
- Time-activated load—to allow you to load software into a device in off-peak hours unattended.

– Protocol Enhancements:

- AppleTalk
  - Support for AppleTalk network management applications.
- APPN
  - Support for APPN Branch Extender, which decreases the overall size of a topology database.
  - Improved management of topology database by removing database records for resources that are no longer active in the network.
  - Support for an APPN Frame Relay port that uses BAN.
  - Support for an implicit focal point, which will allow you to specify a node with network management responsibilities on the device. The device uses this focal point when it cannot establish a session with an explicit focal point.
  - The ability to configure the size of the held alert queue.
- ATM:
  - RFC 1483 bridging over ATM PVCs and SVCs.
  - Shared RFC 1483 PVCs.
  - LAN emulation quality of service.
  - Next Hop Resolution Protocol (NHRP) client.
  - ATM virtual interface.
  - Redundant default IP gateways for emulated LANs.
  - Classical IP (CIP) redundancy.
  - Distributed ARP server support, which provides continuous connectivity across logical IP subnets in the event of an ARP server failure.
  - Support for RFC 1577+ clients.
- BRS
  - The ability to define default circuit definitions for traffic class handling on a Frame Relay interface.
  - Improvements that allow assigning TCP/IP packets to a BRS class and priority based on a TCP/UDP port number, a TCP/UDP port range, or a TCP/UDP socket.
- Frame Relay:
  - Frame Relay now runs on V.25bis.
  - Data compression supported on Frame Relay.

Note: If you are currently using compression on PPP interfaces, you will need to configure compression contexts for compression to work.

- Improved congestion control using BECN, FECN, and CLLM.
- Support for the setting of the discard eligibility (DE) bit in Frame Relay packets.

- IP:
  - Support for RIP Version 2
  - The ability to define up to four static routes for each IP destination host or subnet.
  - The ability to define the same IP subnet on multiple network interfaces.
  - Improved determination of whether a static route will work.
  - Improvements to route filtering and routing policies.
  - OSPF demand circuits for OSPF topology refreshes and Hello messages.
  - PING enhancements.
- IPX:
  - The ability to configure IPX static routes and services to prevent RIP and SAP from activating V.25bis and ISDN demand circuits.
  - The ability to configure a default route to a destination network.
  - PING enhancements.
- ISDN:
  - Native I.430 and I.431 support.
- PPP:
  - Increased bandwidth by increasing the number of links through Bandwidth Allocation Protocol/Bandwidth Allocation Control Protocol (BAP/BACP).
  - Support for Multilink PPP.
  - Enhanced authentication using authentication servers.
  - Support for Encryption Control Protocol (ECP) negotiation using Data Encryption Standard (DES) Cypher Block Chaining (CBC) mode.
  - Support for the Microsoft Point-to-Point Compression (MPPC) protocol.

  Note: If you are currently using compression on PPP interfaces, you will need to configure compression contexts for compression to work.
- SNMP enhancements:
  - Upgrade or addition of the following MIBs:
    - Appletalk MIB
    - APPN family of MIBs

| – ATM
| – BRS MIB
| – Ethernet
| – HPR
| – Interfaces MIB
| – IP over ATM MIB
| – IPX MIB
| – ISDN MIB
| – LAN Emulation Client
| – NHRP MIB
| – PPP MIB.

| • New trap support

| – FR traps for the receipt of FECN, BECN, and CLLM.

| - WAN Reroute enhancements:

| • Allows traffic to flow from a primary circuit to an alternate circuit when a capacity threshold is exceeded.

| • Allows a dial circuit interface to be defined as the primary interface for WAN Reroute purposes.

| - X.25 enhancements:

| • New change commands for XTP.

| • **Clarifications and corrections**

| The technical changes and additions are indicated by a vertical line (|) to the left of the change.

Chapter 1. Getting Started (Introduction to the User Interface)

This chapter shows you how to get started with using the following components related to the IBM 2210 Nways Multiprotocol Router (2210) and the Multiprotocol Routing Services:

- Router console terminals
- Router software (Multiprotocol Routing Services)
- Router software user interface
- Protocol software - configuring and monitoring
- Network interfaces - configuring and monitoring

The information in this chapter is divided into the following sections:

Before You Begin

Before you begin, refer to the following checklist to verify that your router is installed correctly.

HAVE YOU...

- Installed all necessary hardware?
- Connected the console terminal (video terminal) to the router?

  Attention: If you are using a service port-attached terminal to configure or monitor your IBM 2210 and your service terminal is unreadable, you need to change some parameters in your configuration. (See "Service Terminal Display Unreadable" in IBM 2210 Nways Multiprotocol Router Service and Maintenance Manual.)
- Connected your router to the network using the correct network interfaces and cables?
- Run all necessary hardware diagnostics?

For more information on any of these procedures, refer to the *IBM 2210 Nways Multiprotocol Router Installation and Initial Configuration Guide*

Migrating to Release 2.0

Refer to the chapter entitled “Migrating to a New Code Level” in the Maintenance Guide for information about migrating to a new code level.

Using Local and Remote Router Consoles

The router console lets you use the router user interface to monitor and change the function of the router’s networking software (Multiprotocol Routing Services). The router supports local and remote consoles.

Local Consoles

Local consoles are either directly connected by an EIA 232 (RS-232) cable, or connected via modems to the router. You may need to use a local console during the initial software installation. After the initial setup connection, you can connect via Telnet, as long as IP forwarding has been enabled. (Refer to *Protocol Configuration and Monitoring Reference* for more information on enabling IP forwarding.)

When the configured router is started for the first time, a boot message appears on the screen, followed by the OPerator's CONsole or OPCON prompt (*). The * prompt indicates that the router is ready to accept OPCON commands.

Your Multiprotocol Routing Services software may have been pre-configured at the factory. If it was, you do not need to use a local console to perform initial configuration. If, however, your Multiprotocol Routing Services was not pre-configured at the factory, you

Important: Garbage, random characters, reverse question marks, or the inability to connect your terminal to the IBM 2210 service port can have many causes. The following list contains some of those causes:

- The most common cause of garbage or random characters on the service console is that the baud rate is not synchronized with the IBM 2210.

  If the 2210 is set to a specific baud rate, the terminal or terminal emulator must be set to the same baud rate.

  If the IBM 2210 is set to autobaud (this is the default), press the terminal `break` key sequence and press **Enter**.

  A typical break key sequence for PC terminal emulators is `Alt-b` (refer to the terminal emulator documentation). Most ASCII terminals have a `Break` key (often used in conjunction with the `Ctrl` key).
- Defective terminal or device (ac) grounds.
- Defective, incorrectly shielded, or incorrectly grounded EIA 232 (RS-232) cable between the terminal and the IBM 2210.
- Defective terminal or terminal emulator.
- Defective IBM 2210 system board.
- High ambient electromagnetic interference (EMI) levels.
- Power line disturbances.

(See “Service Terminal Display Unreadable” in the *IBM 2210 Nways Multiprotocol Router Service and Maintenance Manual*.)

Once the 2210 is initially configured, you will not need a local console for router operation, as long as IP forwarding is enabled.

The router software automatically handles console activity. When upgrading the software, you might have to use the local console. For information on attaching and configuring local consoles, refer to the *IBM 2210 Nways Multiprotocol Router Installation and Initial Configuration Guide*

Remote Consoles

Remote consoles attach to the router using a standard remote terminal protocol. Remote consoles provide the same function as local consoles, except that a local console must be used for initial configuration if your IBM 2210 was not pre-configured at the factory.

Telnet Connections

The router supports both Telnet Client and Server. The remote console on the router acts as a Telnet server. The router acts as a Telnet client when connecting from the router to either another router or a host using the **telnet** command in the OPCON (*) process.

Remote Login Names and Passwords

During a remote login, the router prompts you for a login name and password. You can display the login name when logged in to the router from a remote console by using a router **status** command.

Use the **set password** command to supply a password for the router. The password, user-configurable for each router, controls access to the router. You may also configure a password for users of a local console.

Note: If you do not enter a login name and valid password within one minute of the initial prompt, or if you enter an incorrect password three times in succession, the router drops the Telnet connection.

Multiple users with login permissions may also be added using the **add user** command. See Chapter 3, “The CONFIG Process and Commands” on page 3-1 for more details on the **add user** commands.

Logging In Remotely or Locally

Logging in to a local console is the same as logging in to a remote console except that you must connect to the router by starting Telnet on your host system. To log in remotely, begin at step 1. To log in locally, begin at step 3 on page 1-4.

To log in from a remote console:

1. Connect to the router by starting Telnet on your host system. Your host system is the system to which remote terminals are connected.
2. Supply the router’s name or Internet Protocol (IP) address.

   To use router names, your network must have a name server. Issue either the router name or the IP address as shown in the following example:

   ```
   % telnet brandenburg
   ```

or

```
% telnet 128.185.132.43
```

The router supports both Telnet Client and Server. The remote console on the router acts as a Telnet server. The router acts as a Telnet client when connecting from the router to either another router or a host using the **telnet** command.

At this point, it makes no difference whether you have logged in remotely or locally.

3. If you are prompted, enter your login name and password.

```
login:
Password:
```

It is possible that there is a login and no password. The password controls access to the router. If a password has not been set, press the **Enter** key at the `Password:` prompt. Logins are not set automatically. For security, you can set up user names and passwords using the **add user** command in the CONFIG process. For additional information, see the **add user** configuration command on page “Add” on page 3-12 Remember to reload or restart to activate their use.

Note: If you do not enter a login name and valid password within one minute of the initial prompt, or if you enter an incorrect password three times in succession, the router drops the Telnet connection.

4. Press the **Enter** key to display the main prompt asterisk (*).

You may have to press the **Enter** key more than once or press **Ctrl** **P** to obtain the * prompt.

Once at this level, you can begin to enter commands from the keyboard. Press the **backspace** key to delete the last character typed in on the command line. Press the **Delete** key or **Ctrl** **U** to delete the whole command line entry so that you can re-enter a command. See “Command History for GWCON and CONFIG Command Line” on page 1-22 for information on how to access previously entered commands.

Once at the main router prompt, you can begin entering commands from the keyboard. (See “Discussing the User Interface System” on page 1-7 for more information on using the router user interface.) To exit the router, return to the main router prompt (*) and close the Telnet connection by typing **logout**. For example:

```
IP Config> exit
Config> Ctrl P
* logout

%
```

You can also use local Telnet commands on your Telnet client to close the Telnet connection.

Executing a Command

When typing a command, remember the following:

- Type only enough sequential letters of the command to make it unique among available commands. For example, to execute the **reload** command you must enter **rel** as a minimum.
- Commands are not case-sensitive.

- Sometimes, only the first letter of the command (and subsequent options) is required to execute the command. For example, typing **s** at the * prompt followed by pressing the **Enter** key causes the **status** command be executed.

  Note: If you use a VT100 terminal, do not press the **Backspace** key, because it inserts invisible characters. Use the **Delete** key.

Connecting to a Process

When you start the router, the console displays a boot message. The OPCON prompt (*) then appears on the screen indicating that you are in the OPCON process and you can begin entering OPCON commands. This is the command prompt from which you communicate with different processes.

To connect your console to a process:

1. Find out the process ID (PID) number of a process by entering the **status** command at the * prompt.

   The **status** command displays information about the router processes, such as the process IDs (PIDs), process names and status of the process. Issuing the **status** command is shown in the following example:

```
* status
Pid  Name      Status TTY  Comments
1    COpCon    RDY    TTY0
2    Monitr    DET     --
3    Tasker    RDY     --
4    MOSDDT    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    Ezystrt   IDL     --
8    ROpCon    IDL    TTY1 128.185.210.125
9    ROpCon    IDL    TTY2
10   CES3      IDL     --
11   TOUT      IDL     --
12   L2S3      RDY     --
13   L3L2      RDY     --
14   LLL2      RDY     --
15   S3CE      RDY     --
```

2. Use the **talk *pid*** command, where *pid* is the number of the process to which you want to connect. (For more information about these and other OPCON commands, refer to Chapter 2, “The OPCON Process and Commands” on page 2-1.)

   Note: All the processes listed do not have a user interface (for example, the **talk 3** process). The **talk 4** command is for use by the software specialist.

Identifying Prompts

Each process uses a different prompt. You can tell which process your console is connected to by looking at the prompt. (If the prompt does not appear when you enter the **talk** *pid* command, press the **Return** key a few times.)

The following list shows the prompts for the three main processes:

Process	Prompt
OPCON	*
GWCON	+
CONFIG	Config>

At the prompt level, you can begin to enter commands from the keyboard. Use the **Backspace** key to delete the last character typed in on the command line. Use **Ctrl** **U** to delete the whole command line entry so that you can re-enter a command. See “Command History for GWCON and CONFIG Command Line” on page 1-22 for information on how to access previously entered commands.

Getting Help

At any of the prompts just described, you can obtain help in the form of a listing of the commands available at that prompt level. This is done by typing **?** (the **help** command), and then pressing the **Return** key. Use **?** to list the commands that are available from the current prompt level. You can usually enter a **?** after a specific command name to list its options. For example, the following appears if you enter **?** at the * prompt:

```
*?

BREAKPOINT
DIVERT output from process
FLUSH output from process
HALT output from process
INTERCEPT character is
LOGOUT
MEMORY statistics
RESTART     RELOAD
STATUS of process(es)
TALK to process
TELNET to IP-Address
```

Getting Back to OPCON

To get back to the OPCON prompt (*), press **Ctrl** **P**. You must always return to OPCON before you can communicate with another process. For example, if you are connected to the GWCON process and you want to connect to the CONFIG process, you must press **Ctrl** **P** to return to OPCON first. The **Ctrl** **P** key combination is called the *intercept character*.

When using third-level processes, such as IP Config or IP shown in Figure 1-1 on page 1-7, use the **exit** command to return to the second level.

Use the intercept character (the default intercept character is **Ctrl** **P**) from a third-level process to return to the * prompt. The next time you use the **talk** command, you will re-enter the third level process. This link goes away when the router is reinitialized.

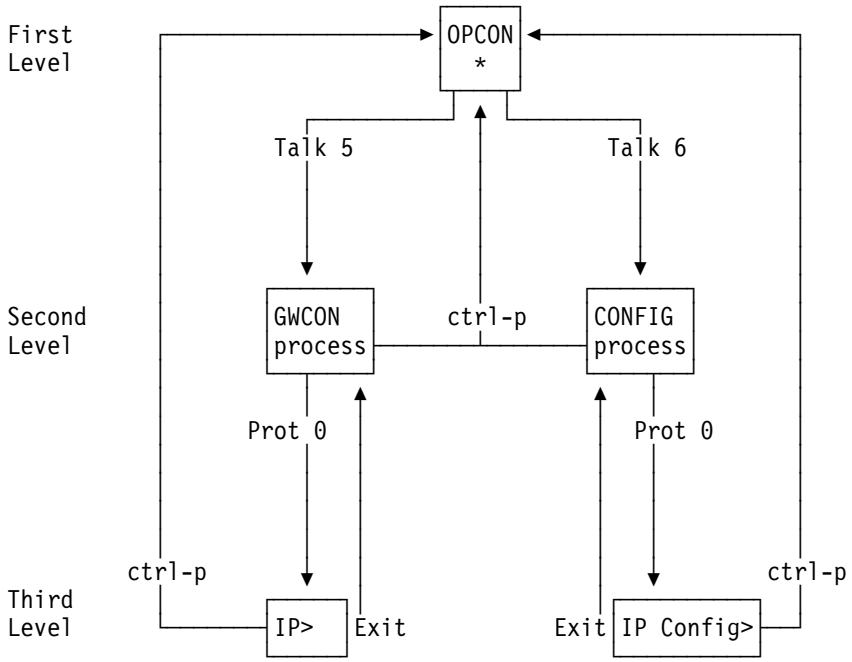


Figure 1-1. Talk and the Intercept Commands

Figure 2-2 on page 2-3 shows the commands in the OPCON structure.

Exiting the Router

Return to the * prompt and close the Telnet connection. For example:

```
IP Config> exit
Config>  Ctrl   P
* logout

%
```

You can also use local Telnet commands on your Telnet client to close the Telnet connection.

Discussing the User Interface System

The user interface to the router software consists of the main menu (process) and several subsidiary menus (processes). The processes you use most often are OPCON, GWCON, CONFIG, and CONFIG-ONLY. These processes allow you to control and monitor the operations of the router.

There are three levels of processes: first, second, and third as you move down the software tree.

Figure 1-2 on page 1-8 shows the processes and how they fit within the structure of the router software.

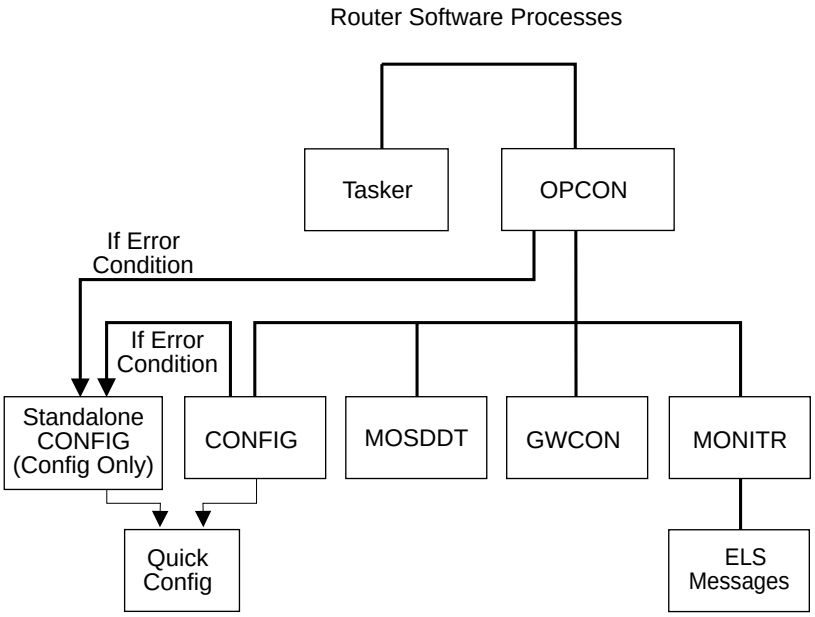


Figure 1-2. Multiprotocol Routing Services

The router software (Multiprotocol Routing Services) is a multi-tasking system that schedules use of the CPU among various processes and hardware devices. The router software:

- Provides timing and memory management, and supports both local and remote operator consoles from which you can view and modify the router's operational parameters.
- Consists of functional modules that include various user interface processes, all network interface drivers, and all protocol forwarders purchased with the router.

Table 1-1 on page 1-9 summarizes the Multiprotocol Routing Services processes and functions on the 2210.

Table 1-1. Router Software Processes

Process	Definition	Prompt
OPCON	Operates as the main operator control program. It provides service for one directly connected console terminal.	Asterisk (*)
ROPCON	Provides OPCON service for two remotely connected console terminals (not shown). Functionally, OPCON and ROPCON are the same.	Asterisk (*)
EZSTRT	The first level of the router's user interface during automatic configuration download. Provides access to second-level processes. Commands that run in OPCON will also run in EasyStart but may or may not be useful. Functionally, OPCON and EasyStart are the same.	`EasyStart>`
GWCON (or CGWCON) (Second Level)	Displays the status and statistics of the router's hardware and software, such as protocols, network interfaces, and event logging. It is on the second level of the router user interface. It provides access to the third-level process, which allow you to monitor configured protocols and features.	Plus sign (+)
CONFIG (Second Level)	Provides on-line control of various configuration parameters, such as network addresses and event logging. It is on the second level of the router user interface. It provides access to third-level processes, which allow you to configure various protocols and features. Quick Config menus are also available under this process.	`Config>`
MONITR	Receives Event Logging System (ELS) messages and messages from the operating system, and displays then on the monitor, according to user-selected filtering criteria. It is on the second level of the router user interface.	None
TASKER	Runs the router's main networking software and performs the router's internetwork data transfer operations. This process is part of the operating system and has no user interface.	None
MOSDDT	Serves as the Micro Operating System (MOS) runtime debugging tool (Dynamic Debugging Tool).	Dollar sign ($)
QUICK CONFIG	Provides a simple, less-detailed way of configuring devices, bridging and routing protocols, and booting records.	None
CONFIG-ONLY (Stand-alone Config)	Provides the same function as the CONFIG process with the addition of the **restart** and **reload** commands. command.	`Config (only)>`

Definition of the First-Level User Interface

The first level of the user interface includes CONFIG-ONLY and OPCON. The CONFIG-ONLY process (which includes (Quick Configuration) allows you to configure the router. Figure 1-3 on page 1-11 illustrates how the router enters either Quick Config or CONFIG-ONLY. The OPCON process allows you to set up communication between users and the other router processes. If you are using a remote console, the name of the process handling your console is ROPCON instead of OPCON, but the operation is identical.

CONFIG-ONLY Process

Configuration Only, or CONFIG-ONLY, allows you to take the router off line and reconfigure its operating parameters.

The commands available are the same as CONFIG at the `Config>` prompt; however, no other processes are running while the router is in CONFIG-ONLY mode.

The CONFIG-ONLY mode uses commands identical to the CONFIG process, with the addition of the **restart** and **reload** commands. command.

CONFIG-ONLY mode is provided only for getting a subset of configuration commands when a configuration problem causes the router to crash with a *panic*, *check*, *fatal*, or *bughlt*, particularly those relating to failures of memory allocations. CONFIG-ONLY mode should be used only to adjust parameters affecting memory allocations such as routing table sizes, packet sizes, and receive buffer allocations. It should not be used for general router configuration. In CONFIG-ONLY mode, many of the device-related commands are disabled and some may cause a crash.

There are two ways to enter CONFIG-ONLY mode:

Operational Failure: The router encounters a problem during initialization and automatically comes up in CONFIG-ONLY mode. Any of the following situations will cause the router to enter into CONFIG-ONLY mode:

- An unsupported device is in the software load
- One or more of the following configuration errors have occurred during start-up:
  - Static RAM is corrupted
  - All router interface information has been deleted.
  - Incorrect interface configuration information has been entered.

Deliberate entry into CONFIG-ONLY mode: To access CONFIG-ONLY mode deliberately, use the **PROM>bc** command when configuring boot options. See Chapter 5, “Boot Options” on page 5-1 for more detail.

Figure 1-3 on page 1-11 illustrates how the router enters either Stand-alone Configuration (CONFIG-ONLY mode) or Quick Configuration.

Quick Configuration Process

Quick Configuration, or Quick Config, allows you to quickly configure portions of the router without dealing with the specific operating system commands. When you initially load, start, or restart the router with no configuration, you enter Config-Only and you can access Quick Config menus from that process. If the router has

devices configured and the devices do not have any protocols configured, the router automatically starts Config-Only and then enters Quick Config.

You can also enter Quick Config from the CONFIG process using the **qconfig** command.

OPCON

The OPCON process handles the communication between users and the other router processes. If you are using a remote console, the name of the process handling your console is ROPCON instead of OPCON, but the operation is identical.

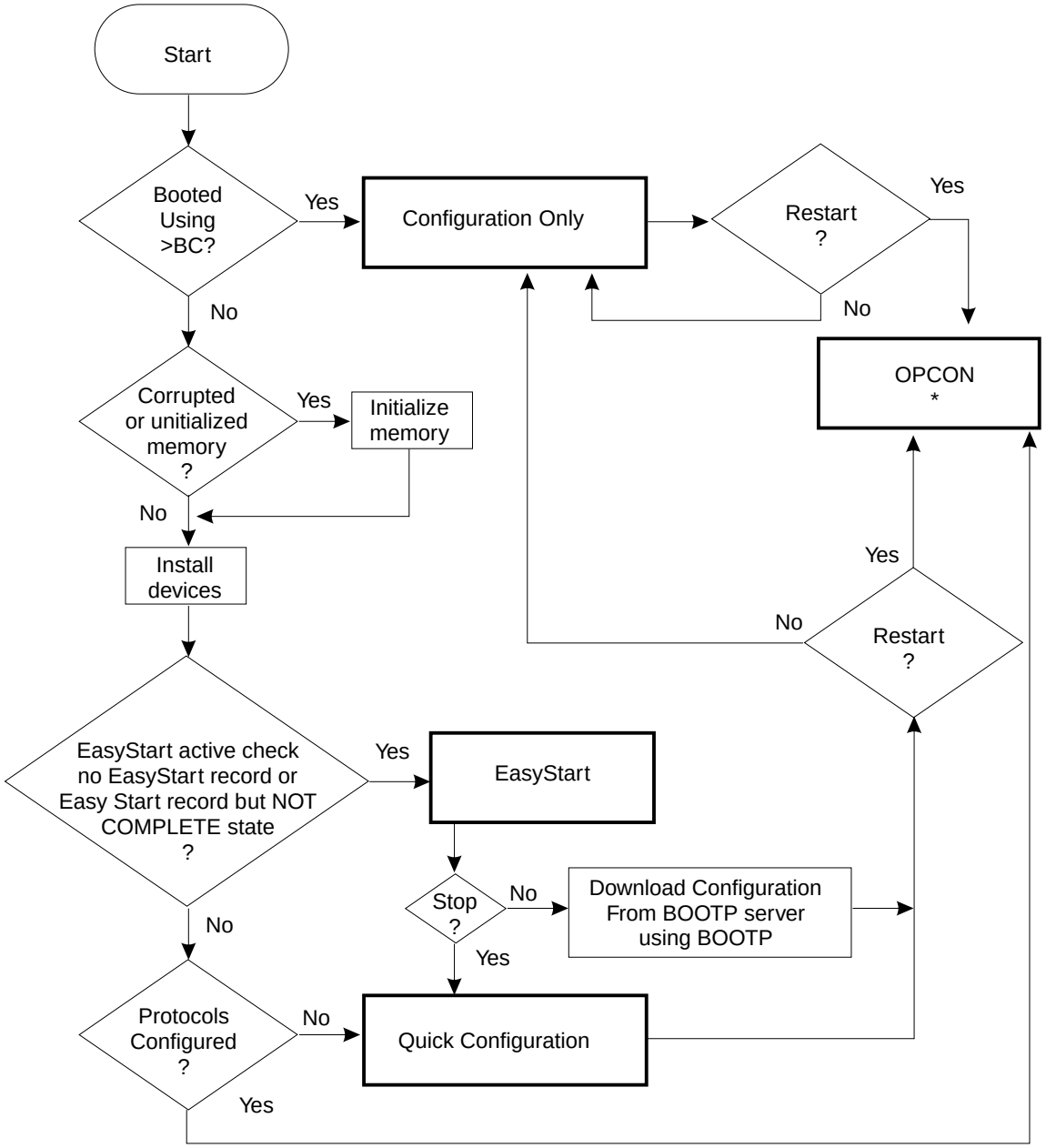


Figure 1-3. Automatic Entry into Configuration Only, EasyStart, OPCON, or Quick Configuration

Refer to Chapter 2, “The OPCON Process and Commands” on page 2-1 for complete details.

Accessing Protocol Configuration and Console Processes

To help you access the configuration and console processes, this section outlines both of these procedures.

All protocols described in the *Protocol Configuration and Monitoring Reference* have commands that are executed in one of the following ways:

- Accessing the protocol *configuration* process to initially configure and enable the protocol, as well as perform later configuration changes.
- Accessing the protocol *console* process to monitor information about each protocol or make temporary configuration changes.

The procedure for accessing these processes is basically the same for all protocols. The next sections describe these procedures.

Accessing the Protocol Configuration Process (CONFIG)

Each protocol configuration process is accessed through the router’s CONFIG process. CONFIG is the second-level process of the router user interface that lets you communicate with third-level processes. Protocol processes are examples of third-level processes.

The CONFIG command interface is made up of levels that are called modes. Protocol configuration command interfaces are modes of the CONFIG interface. Each protocol configuration interface has its own prompt. For example, the prompt for the TCP/IP protocol command interface is `IP config>`.

To access the protocol configuration processes:

1. Enter the CONFIG command process from OPCON and obtain the CONFIG prompt (`Config>`)
2. Enter the desired protocol configuration process (with its own prompt) from the CONFIG prompt using the **protocol** command.

The next sections describe these procedures in more detail.

Entering the CONFIG Process

To enter the CONFIG command process from OPCON and obtain the CONFIG prompt:

1. At the OPCON prompt, enter the **status** command to find the PID for CONFIG. (See page 1-5 for sample output of the **status** command.)
2. Enter the OPCON **talk** command and the PID for CONFIG. The PID for CONFIG is 6.

```
* talk 6
```

The console displays the CONFIG prompt (`Config>`). If the prompt does not appear, press the **Return** key again.

Quick Configuration Process: Quick Configuration, or Quick Config, allows you to quickly configure portions of the router without dealing with the specific operating system commands. You enter the Quick Config menus from the CONFIG process using the **qconfig** command.

Entering the Desired Protocol Configuration Process

To enter the desired protocol configuration process from the CONFIG prompt:

1. At the CONFIG prompt, enter the **list configuration** command to see the numbers and names of the protocols purchased in your copy of the software. See 3-27 for sample output of the **list configuration** command.
2. From the `Config>` prompt, enter the **protocol** command with the number or short name (for example, IP, IPX, and ARP) of the protocol you want to configure. The protocol number and short name is obtained from the **list configuration** command display. In the following example, the command has been entered for accessing the IP protocol configuration process:

   ```
   Config> protocol IP
   ```

   or

   ```
   Config> protocol 0
   ```

   The protocol configuration prompt then displays on the console. The following example shows the IP protocol configuration prompt:

   ```
   IP config>
   ```

You can now begin entering the protocol's configuration commands. See the corresponding protocol section of the *Protocol Configuration and Monitoring Reference* for more information on specific protocol configuration commands.

In summary, the **protocol** command lets you enter the configuration process for the protocol software installed in your router. The **protocol** command enters a protocol's command process. After entering the protocol command, the prompt of the specified protocol appears. From the prompt, you can enter commands specific to that protocol.

Exiting the Protocol Configuration Process

To exit the protocol configuration process:

1. Return to the CONFIG process by entering the protocol **exit** command. For example:

   ```
   IP config> exit
   ```

2. Return to the OPCON process by entering the OPCON intercept character (**Ctrl** **P**). For example:

   ```
   Config> Ctrl P
   ```

Reloading the Router

Changes that you make to the protocol parameters through CONFIG do not take effect until you reload the router software.

Notes:

1. For the 1Sx, 1Ux, 14x, and 24x models, you must enter the **write** command to save the changes in the device's flash memory.
2. For all other models, the changes you make through CONFIG are retained in a configuration database in flash memory. They are retained during power-downs and are recalled when you restart the router.

To restart the router, enter the OPCON **restart** command. For example:

```
* restart
```

Are you sure you want to restart the router? (Yes or No): **yes**

Accessing the Protocol Console (Monitoring) Process, GWCON

To view information about the protocols or to change parameters at the console, you must access and use the protocol console process. Protocol console command interfaces are modes of the GWCON interface. Within the GWCON mode, each protocol console interface has its own prompt. For example, the prompt for the TCP/IP protocol is `IP>`.

To access the protocol console processes:

1. Enter the GWCON command process from OPCON and obtain the GWCON prompt.
2. Enter the desired protocol console process from the GWCON (+) prompt using the **protocol** command.

   For more information on using the **protocol** command, see "Protocol" on page 6-14.

The next sections describe these procedures in more detail.

Entering the GWCON Command Process

To enter the GWCON process from OPCON and obtain the GWCON prompt:

1. At the OPCON (*) prompt, enter the **status** command to find the PID for GWCON. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the OPCON **talk** command and the PID for GWCON. For example:

   ```
   * talk 5
   ```

   The GWCON prompt (+) then displays on the console. If the prompt does not appear, press **Return** again.

Entering a Protocol Console Process

To enter a protocol console process from the GWCON prompt:

1. At the GWCON prompt, enter the **configuration** command to see the protocols and networks configured for the router. For example:

```
+configuration

Portable M68360 C Gateway BENNY S/N 207
Multiprotocol Routing Services
5765-B86 Feature 5xxx V1 R1.0 PTF 0 RPQ 0
Boot ROM version  1.10     Watchdog timer enabled    Auto-boot enabled
Time: 13:43:04    Thursday     March 9, 1995       Console baud rate: 9600

Num Name  Protocol
0   IP    DOD-IP
3   ARP   Address Resolution
7   IPX   Netware IPX
11  SNMP  Simple Network Management Protocol
12  OSPF  Open SPF-Based Routing Protocol
23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge
26  DLS   Data Link Switching

Num Name  Feature
1   BRS   Bandwidth Reservation
2   MCF   MAC Filtering

3 Networks:
Net Interface  MAC/Data-Link         Hardware            State
0   TKR/0      Token-Ring/802.5      IBM Token-Ring      Up
1   FR/0       Frame Relay           SCC Serial Line     Down
2   PPP/0      Point to Point        SCC Serial Line     Up
```

2. Enter the GWCON **protocol** command with the protocol number or short name of the desired protocol displayed in the configuration information.

   In the following example, the command has been entered for accessing the IP protocol console process.

   ```
   + protocol 0
   ```

   or

   ```
   + protocol IP
   ```

   The protocol console prompt then displays on the console. This example shows the IP protocol console prompt:

   ```
   IP>
   ```

You can now begin entering the protocol's commands. See the corresponding protocol section of the *Protocol Configuration and Monitoring Reference* for more information on specific protocol console commands.

Exiting the Protocol Console Process

To exit the protocol console process and return to the OPCON process:

1. Return to the GWCON process by entering the protocol **exit** command. For example:

   ```
   IP> exit
   ```

2. Return to the OPCON process by entering the OPCON intercept character (**Ctrl** **P**). For example:

   ```
   + Ctrl  P
   ```

Protocol Names and Numbers

Table 1-2 lists the numbers that you enter with the **protocol** command to access a specific protocol configuration or console process.

Table 1-2. Protocol Numbers and Names

Protocol Number	Protocol Short Name	Accesses the following protocol process
0	IP	IP (Internet Protocol)
3	ARP	ARP (Address Resolution Protocol)
4	DN	DNA Phase IV
6	VIN	Banyan VINES
7	IPX	IPX (Novell NetWare Internetwork Packet Exchange)
8	OSI	ISO CLNP/ESIS/ISIS
9	DVM	Distance Vector Multicast Routing Protocol
10	BGP	BGP (Border Gateway Protocol)
11	SNMP	SNMP (Simple Network Management Protocol)
12	OSPF	OSPF (Open Shortest Path First)
20	SDLC	SDLC Relay
22	AP2	AppleTalk Phase 2
23	ASRT	Adaptive Source Routing Transparent Bridge
24	HST	TCP/IP Host Services
25	LNM	LAN Network Manager
26	DLS	Data Link Switching
27	XTP	X.25 Transport Protocol
28	APPN HPR	APPN High Performance Routing
30	APPN ISR	APPN Intermediate Session Routing

Accessing Feature Configuration and Console Processes

To help you access the Multiprotocol Routing Services feature configuration and console processes, this section outlines both of these procedures.

All Multiprotocol Routing Services features described in this guide have commands that are executed in one of the following ways:

- Accessing the *configuration* process to initially configure and enable the feature as well as perform later configuration changes.
- Accessing the *console* process to monitor information about each feature or make temporary configuration changes.

The procedure for accessing these processes is basically the same for all features. The next sections describe these procedures.

Accessing the Feature Processes

Use the **feature** command from the CONFIG process to access configuration commands for specific Multiprotocol Routing Services features outside of the protocol and network interface configuration processes.

Use the **feature** command from the GWCON process to access console commands for specific features that are outside of the protocol and network interface console processes.

Enter a question mark after the **feature** command to display a listing of the features available for your software release. For example:

```
Config> feature ?

WRS
BRS
MCF

Feature name or number [1] ?
```

To access a particular feature's configuration or console prompt, enter the **feature** command at the Config> or + (GWCON) prompt, respectively, followed by the feature number or short name. For example:

```
Config> feature mcf

MAC filtering user configuration

Filter Config>
```

Table 3-4 on page 3-26 lists the available feature numbers and names.

Once you access the configuration or console prompt for a feature, you can begin entering specific commands for the feature. To return to the previous prompt level, enter the **exit** command at the feature's prompt.

Accessing Network Interface Configuration and Console Processes

This section describes how to get started with accessing the network interface configuration and console processes. Accessing these processes lets you change and monitor software-configurable parameters for network interfaces used in your router.

Although this manual concerns itself primarily with configuration tasks, there might be some point during configuration when you want to display some interface statistics and information. To help you access both the configuration *and* console processes, this manual outlines both procedures. The information presented in the next sections includes:

- "Accessing the Network Interface Configuration Process" on page 1-18
- "Accessing the Network Interface Console Process" on page 1-21

Accessing the Network Interface Configuration Process

Use the following procedure to access the router's configuration process. This process gives you access to a specific interface's *configuration* process.

1. At the OPCON prompt (*), enter the **status** command to find the PID for CONFIG. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the OPCON **talk** command and the PID for CONFIG. (For more detail on this command, refer to Chapter 2, "The OPCON Process and Commands" on page 2-1.) For example:

   ```
   * talk 6
   ```

   After you enter the talk 6 command, the CONFIG prompt (Config>) displays on the console. If the prompt does not appear when you first enter **CONFIG**, press **Return** again.

   Use the **add device** command to create a network interface. The add device command automatically assigns the interface number and supports the following types of devices:

   Notes:

   a. Interfaces are automatically created for the base ports and ports on an adapter inserted into the feature slot for those models that have a feature slot, so you only need to use the **add device** command to create virtual interfaces. The examples below show the types of virtual interfaces that can be added.

   b. When interfaces are created for serial adapters or dial circuits, the default data-link type is PPP. However, you can use the **set data-link** command to change the data-link type. Refer to Table 1-3 on page 1-20 for the data-link types supported on serial ports and dial circuits, and to the description of the **set data-link** command on page 3-33.

   Enter **add device ?** to get a list of the supported device types.

   a. Dial circuits

      The following example adds a dial circuit interface:

      ```
      Config> add device dial-circuit
      Adding device as interface 8
      Defaulting Data-link protocol to PPP
      Use "set data-link" command to change the data-link protocol
      Use "net 8" command to configure circuit parameters
      ```

   b. The following example adds a dial-in circuit:

      Note: The dial-in device type is only supported if the software load includes the DIALs feature.

      ```
      Config>add device dial-in
      Adding device as interface 5
      Defaulting Data-link protocol to PPP
      Use "net 5" command to configure circuit parameters
      Base net for this circuit [0]? 4
      ```

   c. The following example adds a dial-out circuit:

      Note: The dial-out device type is only supported if the software load includes the DIALs feature.

```
Config>add device dial-out*
Adding device as interface 6*
Defaulting Data-link protocol to Dial-out*
Use "net 6" command to configure circuit parameters*
Base net for this circuit [0]? 4
```

d. The following example adds a multilink PPP interface:

```
Config>add device multilink-ppp
Adding device as interface 7
Defaulting Data-link protocol to PPP
Use "net 7" command to configure circuit parameters
```

3. At the `Config >` prompt, enter the **list devices** command to display the network interface numbers for which the router is currently configured, as follows:

```
Config> list devices

Ifc 0 Ethernet                      CSR  81600, CSR2  80C00, vector 94
Ifc 1 WAN X.25                      CSR  81620, CSR2  80D00, vector 93
Ifc 2 WAN X.25                      CSR  81640, CSR2  80E00, vector 92
Ifc 3 WAN PPP                       CSR 381620, CSR2 380D00, vector 125
Ifc 4 WAN Frame Relay               CSR 381640, CSR2 380E00, vector 124
Ifc 5 Token Ring                    CSR 600000, vector 95
```

4. Record the interface numbers.
5. Enter the CONFIG **network** command and the number of the interface you want to configure. For example:

```
Config> network 1
```

The appropriate configuration prompt (such as `TKR Config>` for token-ring),

Note: Not all network interfaces are user-configurable. For interfaces that cannot be configured, you receive the message:

```
That network is not configurable
```

Displaying the Interface Configuration

From the same interface configuration prompts, you can list configuration information specific to that selected interface by using the **list** command. For example:

```
TKR Config> list

Token-Ring configuration:

PACKET SIZE (INFO FIELD): 4472
Speed:                    16 Mb/sec
Media:                    Shielded

RIF Aging Timer:          120
Source Routing:           Enabled
MAC Address:              000000000000
```

Configuring the Network Interface

Refer to the specific chapters in this guide for complete information on configuring your IBM 2210's network interfaces.

Table 1-3 on page 1-20 lists network architectures and the supported interfaces for each architecture.

Table 1-3. Network Architecture and the Supported Interfaces

Network Architecture	Supported Interfaces
ATM	Dual Port Serial Interface (25 Mbps) for IBM 2210
802.5 Token-Ring	IBM 2210 Token-Ring 4/16 Interface
Ethernet	IBM 2210 Ethernet Interface
ISDN	Serial Interfaces for IBM 2210 as follows: Basic Rate Interface (BRI) T1/J1 PRI Interface E1 PRI Interface
Point-to-Point	Serial Interface for IBM 2210, dial circuit interface; supported on 4-port and 8-port WAN concentration adapters
Frame Relay	Serial Interface for IBM 2210, dial circuit interface; supported on 4-port and 8-port WAN concentration adapters
X.25	Serial Interface for IBM 2210; supported on 4-port and 8-port WAN concentration adapters
SDLC Relay	Serial Interface for IBM 2210; supported on 4-port and 8-port WAN concentration adapters
SDLC	Serial Interface for IBM 2210; supported on 4-port and 8-port WAN concentration adapters
V.25bis	Serial Interface for IBM 2210; supported on 4-port and 8-port WAN concentration adapters
V.34	Serial Interface for IBM 2210; supported on 4-port and 8-port WAN concentration adapter
Dial-Out	Supports DIALs and telnet dial-out over V.34 base interfaces
Dial-In	A PPP dial circuit interface that has configuration parameters defaulted to support DIALs
Multilink PPP (MP)	Not supported on physical interfaces, only on a virtual interface

Notes:

1. PPP dial circuit interfaces can use an ISDN, V.25bis, or a V.34 network as the base network interface.
2. FR dial circuit interfaces can use an ISDN or a V.25bis network as the base network interface.
3. Dial-Out circuit interfaces use a V.34 network as the base network interface.
4. Dial-In circuit interfaces can use an ISDN or V.34 network as the base network interface.

Exiting the Interface Configuration Process

After you have configured the interface information, exit the interface configuration process by using the following procedure:

1. Return to the CONFIG process by entering the **exit** command. For example:

```
TKR Config> exit
```

2. Return to the OPCON process by entering the OPCON intercept character. The default intercept character is **Ctrl** **P** . (See “Intercept” on page 2-6 for more information.) For example:

   ```
   Config> Ctrl  P
   ```

Restarting the Router

Whenever you change a user-configurable parameter, you must restart the router for the change to take effect. To do so, enter the OPCON **restart** command. For example:

```
* restart

Are you sure you want to restart the gateway? (Yes or No): yes
```

Accessing the Network Interface Console Process

To monitor information related to a specific interface, access the interface console process by using the following procedure:

1. Enter the **status** command to find the PID for GWCON. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the OPCON **talk** command and the PID for GWCON. For example:

   ```
   * talk 5
   ```

3. The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter GWCON, press **Return** again.
4. At the GWCON prompt, enter the **configuration** command to see the protocols and networks for which the router is configured. For example:

   ```
   + configuration

     Portable M68360 C Gateway [not configured] S/N 207
   Multiprotocol Routing Services
   5765-B86 Feature 5xxx V1 R1.0 PTF 0 RPQ 0
   Boot ROM version  1.20     Watchdog timer enabled    Auto-boot enabled

   Time: 13:34:56    Thursday     March 9, 1995       Console baud rate: 9600

   Num Name  Protocol
   0   IP    DOD-IP
   3   ARP   Address Resolution
   11  SNMP  Simple Network Management Protocol
   12  OSPF  Open SPF-Based Routing Protocol
   23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge

   Num Name  Feature
   1   BRS   Bandwidth Reservation
   2   MCF   MAC Filtering

   3 Networks:
   Net Interface  MAC/Data-Link         Hardware         State
   0   Eth/0      Ethernet/IEEE 802.3   SCC Ethernet     Up
   1   PPP/0      Point to Point        SCC Serial Line  Up
   2   PPP/1      Point to Point        SCC Serial Line  UP
   ```

5. Enter the GWCON **network** command and the number of the interface you want to monitor. For example:

```
+ network 2
```

In this example, the X.25 console prompt is displayed on the console. You can then view information about the X.25 interface by entering the X.25 console commands.

```
X.25>
```

Monitoring the Network Interface

Refer to the specific chapters in this manual for complete information on monitoring your 2210's network interfaces.

Exiting the Interface Console Process

To exit the interface console process and return to the OPCON process:

1. Return to the GWCON process by entering the **exit** command. For example:

   ```
   X.25> exit
   ```

2. Return to the OPCON process by entering the OPCON intercept character. (The default intercept character is **Ctrl** **P**.) For example:

   + **Ctrl** **P**

Command History for GWCON and CONFIG Command Line

The Command History contains up to the last 50 commands entered by the user in GWCON (Talk 5) or CONFIG (Talk 6) command line menus.

Backward and Forward retrieve keys can be used to recall previously entered commands. In addition, a facility is provided to enable the advanced user to repeat a particular series of commands.

Repeating a Command in the Command History

By hitting **Ctrl** **B** (BACKWARD) or **Ctrl** **F** (FORWARD) at any command line prompt in a GWCON or CONFIG menu, the current command line is replaced by the previous or next command in the Command History. The Command History is common to both GWCON and CONFIG. That is, a command entered while in a GWCON menu can be retrieved from within CONFIG and a command entered while in a CONFIG menu can be retrieved from within GWCON.

The Command History contains the most recently entered commands, up to a maximum of the last 50 commands. If only three commands have been entered since a restart, hitting **Ctrl** **F** or **Ctrl** **B** circles through only those three commands. If no commands have been entered thus far, **Ctrl** **F** or **Ctrl** **B** results in a "bell," the same bell you see when trying to backspace beyond the beginning of a line of text.

Note: A command aborted by entering **Ctrl** **U** will not be entered into the Command History.

To enter two similar commands:

```
ELS>display sub les
ELS>display sub lec
```

Enter:

ELS>display sub les **enter**

ELS>Enter **Ctrl** **B** for BACKWARD, and the current line is
replaced with-
ELS>display sub les
Enter **backspace** and replace "s" with "c" to get
ELS>display sub lec
then press **enter**

Repeating a Series of Commands in the Command History

There is an additional feature for advanced users to facilitate repeating a particular series of GWCON or CONFIG commands. C1, C2,...,Cn in the Command History is referred to as a *repeat sequence*. This feature may be more convenient than simply using **Ctrl** **B** and **Ctrl** **F** when you must repeat a given task that requires multiple commands. Enter **Ctrl** **R** (REPEAT) to set the start of the *repeat sequence* at command C1. Enter **Ctrl** **N** (NEXT) successively to retrieve the next command(s) in the repeat sequence. Commands are not automatically entered, but are placed on the current command line allowing you to modify or enter the command.

To produce the desired behavior of a repeat sequence, the first command retrieved using the first **Ctrl** **N** (NEXT) depends on the manner in which the start of the repeat sequence was set using **Ctrl** **R** (REPEAT).

Setting the start of the repeat sequence with **Ctrl** **R** can be done in two ways:

1. When C1 is initially entered
2. When C1 is retrieved from the Command History with **Ctrl** **B** or **Ctrl** **F**.

Starting a Repeat Sequence As Commands Are Entered

If you enter **Ctrl** **R** as command C1 is being keyed in, and then enter commands C2, C3... Cn. **Ctrl** **N** will successively bring commands C1, C2, ... Cn, C1, C2, ... Cn, C1, ... to the command line.

In Example 1, the start of the repeat sequence is set as the first command is keyed in. The user knows ahead of time that the same commands to be entered in GWCON need to be repeated in CONFIG.

Example 1

1. As the first command in the sequence is keyed in, use **Ctrl** **R** (REPEAT) to set the start of the repeat sequence.

   ```
   *talk 5
   +event-enter Ctrl R for REPEAT to set the start of the
          repeat sequence-
   +event enter
   ```

2. Continue typing the subsequent commands in the sequence:

   ```
   Event Logging System user console
     ELS>display sub les
     ELS>display sub lec
     ELS>exit
     +
   ```

3. To enter these same commands in CONFIG, press **Ctrl** **P** (the default OPCON intercept character) and go to CONFIG.

```
+-press Ctrl P -
*talk 6
Config>-press Ctrl N for NEXT to retrieve the start of
        this sequence-
Config>event enter
Event Logging System user configuration
ELS config>-press Ctrl N for NEXT to retrieve the next
            command in sequence-
ELS config>display sub les enter
ELS config>-press Ctrl N for NEXT to retrieve the next
            command in sequence-
ELS config>display sub lec enter
ELS config>-press Ctrl N for NEXT to retrieve the next
            command in sequence-
ELS config>exit enter
Config>
```

Starting a Repeat Sequence After All Commands Are Entered

On the other hand, if you first enter C1, C2, ... Cn, and retrieve C1 via **Ctrl** **B** or **Ctrl** **F**. Entering **Ctrl** **R**, entering **Ctrl** **N** successively brings commands C2,..., Cn, C1, C2,..., Cn, C1,...,Cn to the command line (see Example 2). The first occurrence of C1 is bypassed since C1 is already available on the command line at the time it was retrieved, and does not need to be recalled again by the first **Ctrl** **N**.

In Example 2, all the commands are entered and then the first command in the sequence to be repeated is retrieved. A sequence of commands has been entered in GWCON, and the same sequence needs to be repeated in CONFIG.

Example 2

1. Enter the following commands in GWCON:

   ```
   *talk 5
   +event
   Event Logging System user console
   ELS>display sub les
   ELS>display sub lec
   ELS>exit
   +
   ```

2. To enter these same commands in CONFIG, press **Ctrl** **P** (the default OPCON intercept character) and go to CONFIG.

```
+-press Ctrl P -
*talk 6
Config>-press Ctrl B four times to retrieve the start of
        the four command sequence in this example-
Config>event
Config>event-press Ctrl R for REPEAT to set the start of
             the repeat sequence-
 Config>event enter
 Event Logging System user configuration
 ELS config>-press Ctrl N for NEXT to retrieve the next
             command in sequence-
 ELS config>display sub les enter
 ELS config>-press Ctrl N for NEXT to retrieve the next
             command in sequence-
 ELS config>display sub lec enter
 ELS config>-press Ctrl N for NEXT to retrieve the next
             command in sequence-
 ELS config>exit enter
 Config>
```

If the OPCON **intercept** command described in Chapter 2, “The OPCON Process and Commands” on page 2-1 has been used to redefine the OPCON intercept character from the default character Ctrl P to one of the Command History control characters, Ctrl B, Ctrl F, Ctrl R, or Ctrl N, the OPCON intercept character will take priority. For example, if the intercept character has been changed to Ctrl F, then Ctrl F will not retrieve Forward in the Command History, but will instead place the user back at the OPCON prompt (*).

System Security

Multiple users with login permissions can also be added using the **add user** command. See “Configuring User Access” on page 3-7 for details on security issues and for information on the **set password** and **add user** commands.

Chapter 2. The OPCON Process and Commands

This chapter describes the OPCON process and includes the following sections:

What is OPCON?

The Operator Console process (OPCON) is the root-level process of the router software user interface. The main function of OPCON is to control which processes are connected to consoles. OPCON fits into the router software structure as shown in Figure 2-1. Using OPCON commands, you can:

- Manipulate the output from a process
- Change the intercept character
- Display information about router memory usage
- Restart the router software
- Reload the router software (reboot)
- Telnet to other routers or hosts
- Display status information about all router processes
- Communicate with processes at the secondary level
- Escape to the MOS system debugging tool

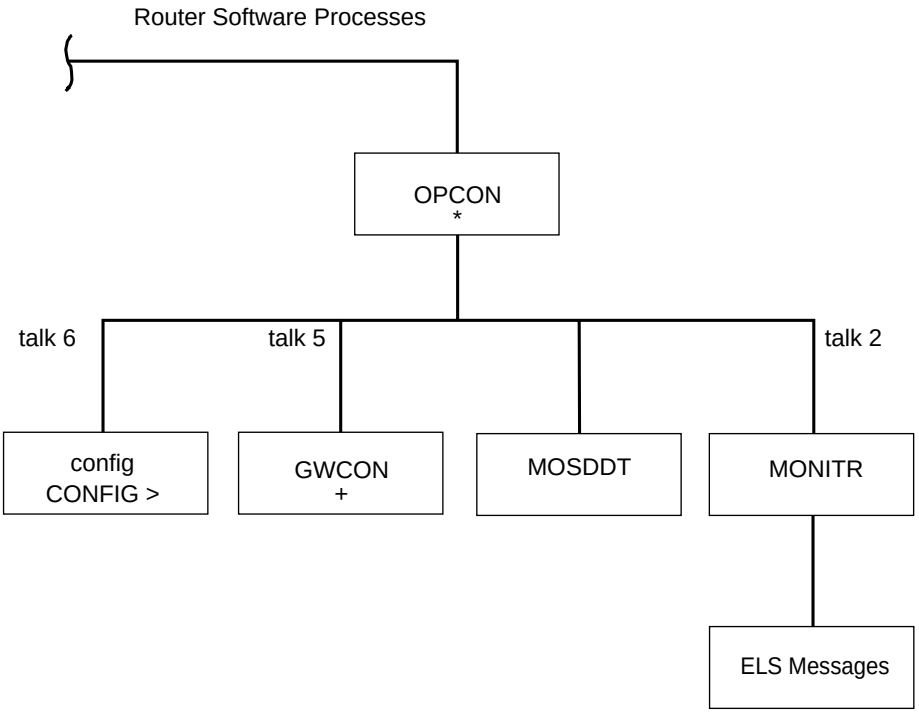


Figure 2-1. OPCON in the Router Software Structure

Accessing the OPCON Process

When you start the router for the first time, a boot message appears on the console. Then the OPCON prompt (*) appears on the console, indicating that you are in the OPCON process and can begin entering OPCON commands.

When you access OPCON from a remote terminal, you access the ROPCON (Remote Operator Console) process. ROPCON and OPCON are functionally the same. If you are in EasyStart, the OPCON prompt appears as the `EasyStart>` prompt instead of the * prompt. OPCON commands run the same with either prompt except that the **pause** and **stop** commands run at the `EasyStart>` prompt.

The OPCON process allows you to configure and monitor all of the router's operating parameters. While in the OPCON process, the router is forwarding data traffic. When the router is booted and enters OPCON, a copyright logo and an asterisk (*) prompt appears on the locally attached console terminal. This is the OPCON (OPerator's CONsole) prompt, the main user interface that allows access to second-level processes.

Some changes to the router's operating parameters made while in OPCON take effect immediately without requiring reinitializing of the router. If the changes do not take effect, use the restart command at the * prompt.

At the * prompt, there is an extensive set of commands that you can enter to check the status of various internal software processes, monitor the performance of the router's interfaces and packet forwarders, and configure various operational parameters.

OPCON Commands

This section describes the OPCON commands. Each command includes a description, syntax requirements, and an example. The OPCON commands are summarized in Table 2-1 on page 2-4. To use them, access the OPCON process and enter the appropriate command at the OPCON prompt (*).

Figure 2-2 on page 2-3 shows the OPCON Command Tree.

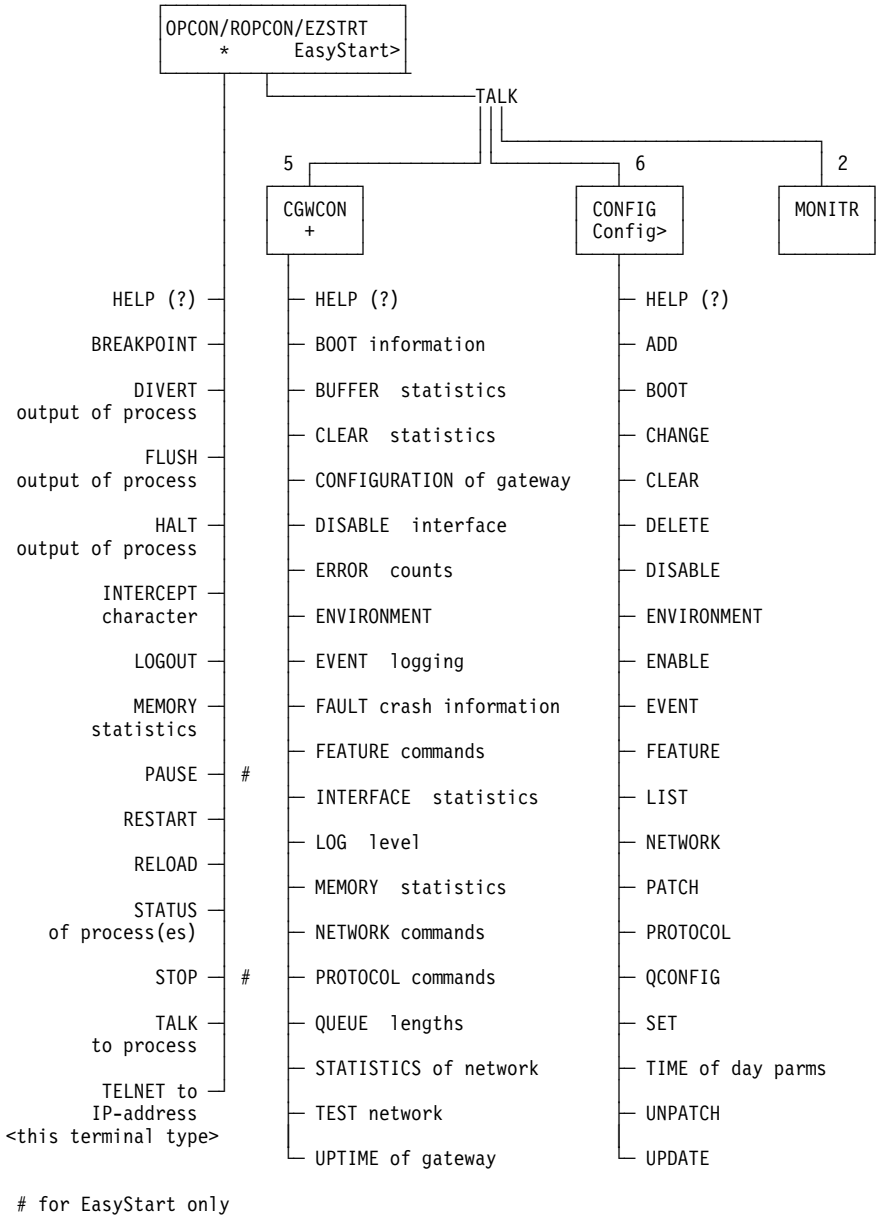


Figure 2-2. OPCON Command Tree

Table 2-1. OPCON Commands

Command	Function
? (Help)	Lists all the OPCON commands.
Breakpoint	Enters the MOS system debugging tool.
Divert	Sends the output from a process to a console or other terminal.
Flush	Discards the output from a process.
Halt	Suspends the output from a process.
Intercept	Sets the OPCON default intercept character.
Logout	Logs out a remote console.
Memory	Reports the router's memory usage.
Pause	Suspends EasyStart (for EasyStart only).
Restart	Restarts (but does not reload) the router software.
Status	Shows information about all router processes.
Stop	Stops EasyStart and enters Config Only mode (for EasyStart only).
Talk	Connects to another router process and enables the use of its commands.
Telnet	Connects to another router.

? (Help)

Use the **?** (**help**) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: ?

```
BREAKPOINT           DIVERT output from process
FLUSH output from process
HALT output from process
INTERCEPT character is
LOGOUT
MEMORY statistics
RESTART           STATUS of process(es)
TALK to process
TELNET to IP Address (this terminal type)
```

Breakpoint

Use the **breakpoint** command to trap information in the MOS system debugging tool, inspect memory, place breakpoints, or obtain a core dump. This command should be used only by software specialists.

If the watchdog timer is on when you invoke this command, the contents of core memory are dumped (if dumping is enabled) when the watchdog timer fires. All routing processes are halted.

The **breakpoint** command must be issued from a local console.

Note: Do not use this command during normal operations because it completely halts operation of the software. If you accidentally enter the **breakpoint** command, quickly press **Esc**, and then **p**.

Syntax: breakpoint

Example: **`breakpoint`**

Divert

Use the **divert** command to send the output from a specified process to a specified terminal. This command allows you to divert the output of several processes to the same terminal to simultaneously view the output. The **divert** command is commonly used to redirect MONITR output messages to a specific terminal. The router allows only certain processes to be redirected.

After entering the command, enter the PID and tty# (number of the output terminal). To obtain these values, use the OPCON status command. The terminal number can be the number of either the local console (tty0) or one of the remote consoles (tty1, tty2). The following example shows Event Logging System messages generated by the MONITR process (2) being sent to a remote console *tty1* (1).

Event messages are displayed immediately even though you may be in the middle of typing a command. The display and keyboard have separate buffers to prevent command confusion. The following example shows the MONITR process connected to TTY1 after executing the **divert 2 1** command. If you want to stop the output, enter **halt 2**. The **halt** command is discussed at “Halt” on page 2-6.

Syntax: divert *pid tty#*

Example: **`divert 2 1`**

```
Copyright Notices:
Copyright IBM Corp. 1994, 1997
MOS Operator Control


* divert 2 1

* status
Pid  Name      Status TTY  Comments
1    COpCon    IOW    TTY0 gzs
2    Monitr    IDL    TTY0
3    Tasker    RDY     --
4    MOSDDT    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    Ezystrt   IDL     --
8    ROpCon    IDL    TTY1
9    ROpCon    RDY    TTY2 jlg@128.185.40.40
10   CES3      IDL    --
11   TOUT      IDL    --
12   L2S3      IDL    --
13   L3L2      IDL    --
14   LLL2      IDL    --
15   S3CE      IDL    --
```

Flush

Use the **flush** command to clear the output buffers of the MONITR process. This command is generally used prior to displaying the contents of the MONITR's FIFO buffer to prevent messages from scrolling off the screen. Accumulated messages are discarded.

The router allows only certain processes to be redirected. To obtain the *pid* and *tty#*, use the OPCON **status** command. As you can see in the following example, after executing the **flush 2** command, the output of the MONITR process is sent to the SNK (it has been flushed).

Syntax: flush *pid*

Example: `flush 2`

```
* status
Pid  Name      Status TTY  Comments
1    COpCon    IOW    TTY0 gzs
2    Monitr    IDL    SNK
3    Tasker    RDY     --
4    MOSDDT    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    Ezystrt   IDL     --
8    ROpCon    IDL    TTY1
9    ROpCon    RDY    TTY2 jlg@128.185.40.40
```

Halt

Use the **halt** command to suspend all subsequent output from a specified process until the **divert**, **flush**, or **talk** OPCON command is issued to the process. The router cannot redirect all processes. **Halt** is the default state for output from a process. To obtain the PID for this command, use the OPCON **status** command. As you can see in the following example, after executing the **halt 2** command, the MONITR process is no longer connected to TTY1. Event messages no longer appear.

Syntax: halt *pid*

Example: `halt 2`

```
* status
Pid  Name      Status TTY  Comments
1    COpCon    IOW    TTY0 gzs
2    Monitr    IDL     --
3    Tasker    RDY     --
4    MOSDDT    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    Ezystrt   IDL     --
8    ROpCon    IDL    TTY1
9    ROpCon    RDY    TTY2 jlg@128.185.40.40
```

Intercept

Use the **intercept** command to change the OPCON intercept character. The intercept character is what you enter from other processes to get back to the OPCON process. The default intercept key combination is **Ctrl** **P**.

The intercept character ***must*** be a control character. Enter the ^ (shift 6) character followed by the letter character you want for the intercept character.

Note: Do not set the intercept character to the return key or to a printable character. If you change the OPCON intercept character from the default, **Ctrl-P**, to one of the Command History control characters, **Ctrl-B**, **Ctrl-F**, **Ctrl-R**, or **Ctrl-N**, the OPCON intercept character will take priority.

For example, if you change the intercept character to **Ctrl-F**, then **Ctrl-F** will not retrieve Forward in the Command History, but will instead return to the OPCON prompt (*). See “Command History for GWCON and CONFIG Command Line” on page 1-22 for information on how to access previously entered GWCON or CONFIG commands.

Syntax: intercept *character*

Example: `intercept ^u`

From this example, you will have an intercept character of **Ctrl** **U**.

Logout

Use the **logout** command to terminate the current session for the user who enters the logout command. If the console login is enabled, this command will require the next user to log in using an authorized userid/password combination. If the console login is not enabled, the OPCON prompt appears again.

Syntax: logout

Example: `logout`

Memory

Use the **memory** command to obtain and display information about the router's global heap memory usage. The display helps you to determine if the router is being utilized efficiently. For an example of memory utilization, see Figure 2-3.

Syntax: memory

Example: `memory`

```
Number of bytes:  Busy = 319544, Idle = 1936, Free = 1592
```

Busy Specifies the number of bytes currently allocated.

Idle Specifies the number of bytes previously allocated but freed and available for reuse.

Free Specifies the number of bytes that were never allocated from the initial free storage area.

Note: The sum of the Idle and Free memory equals the total available heap memory.

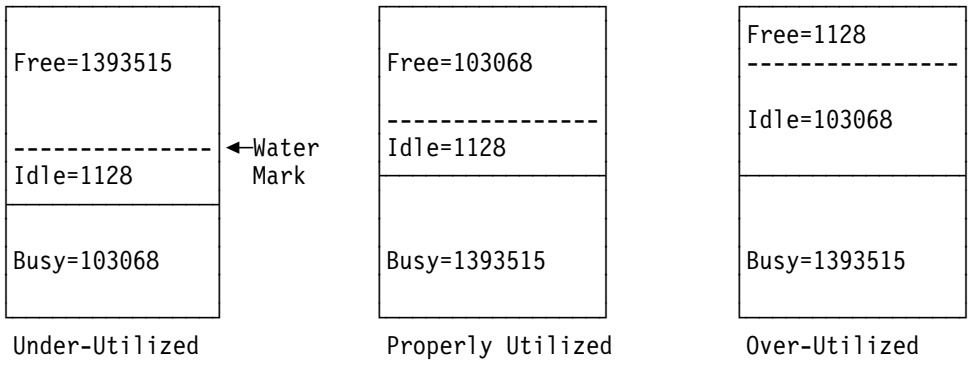


Figure 2-3. Memory Utilization

Pause (EasyStart only)

Use the **pause** command to suspend the EasyStart function. Use this command only when debugging the router. After completing your debugging session, enter the **restart** command to restart the router and resume the EasyStart function. The router will re-enter EasyStart.

Syntax: pause

Example: `pause`

```
Entering EasyStart operation. Type 'stop' to terminate.
ELS messages are automatically displayed in this mode.

EasyStart>

  EZ.001: Starting.
  EZ.007: Waiting up to 6 seconds for devices to pass self-test.
pause
* restart
Are you sure you want to restart the gateway? (Yes or [No]):  yes


Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
Entering EasyStart operation. Type 'stop' to terminate.
ELS messages are automatically displayed in this mode.

EasyStart>

EZ.001: Starting.
  EZ.007: Waiting up to 60 seconds for devices to pass self-test.
BTP.010: net 0, int TKR/0, Sent client request (htype: 6)
BTP.011: net 1, int FR/0, Could not snd client req because: Ifc not up
BTP.011: net 2, int FR/1, Could not snd client req because: Ifc not up
BTP.011: net 3, int FR/2, Could not snd client req because: Ifc not up
```

Restart

Use the **restart** command to reinitialize the software. After you reinitialize the software, a bus reset occurs. This causes the connected network interfaces to self-test, all routing tables to clear, and any packets in the router to drop. Before the restart takes effect, you are prompted to confirm the restart.

Note: If you use this command from a remote console, your Telnet session will be lost because all router processes are being restarted.

Syntax: restart

Example: `restart`

```
Are you sure you want to restart the gateway (Yes or No)? Yes

Copyright Notices:
Copyright IBM Corp. 1994, 1997
MOS Operator Control
*
```

Status

Use the **status** command to display information about all router processes. By entering the PID after the **status** command, you can select to look at the status of only the desired process. The following example shows the total status display.

Syntax: status *pid*

Example: `status`

```
Pid  Name      Status TTY  Comments

1    COpCon    IOW    TTY0
2    Monitr    IDL     --
3    Tasker    RDY     --
4    MOSDDT    DET     --
5    CGWCon    IOW     --
6    Config    IOW    TTY1
7    Ezystrt   IDL     --
8    ROpCon    IOW    TTY1 128.185.46.101
9    ROpCon    RDY    TTY2 128.185.46.104
```

Pid Specifies the PID. This is the process to talk to or from OPCON, or it can be an argument to the STATUS command to request status information about a specific process.

Name Specifies the process name. It usually corresponds to the name of the program that is running in the process.

Status Specifies one of the following:

IDL Specifies that the process is idle and waiting for completion of some external event, such as asynchronous I/O.

RDY Specifies that the process is ready to run and is waiting to use the CPU.

IOW Specifies that the process is waiting for synchronous I/O, usually its expected standard input, to complete.

DET Specifies that the process has output ready to be displayed and it is either waiting to be attached to a display console or waiting to have its output diverted to a specified console.

FZN Specifies that the process is frozen due to an error. This usually means the process is trying to use a device which is faulty or incorrectly configured.

TTYn Specifies the output terminal, if any, to which the process is currently connected.

TTY0	Local console
TTY1 or TTY2	Telnet consoles.
SNK	Process has been flushed.
Two dashes (--)	Process has been halted.

Comments

Specifies the user's login IP address provided during login when a user is logged in using Telnet (ROpCon).

Stop (EasyStart only)

Use the **stop** command to stop the EasyStart function and enter Config-only mode. For information about Config-only mode, see “Config-Only Mode” on page 3-3.

Syntax: stop

Example: **`stop`**

```
EasyStart>  EZ.001: Starting.
  EZ.007: Waiting up to 6 seconds for devices to pass self-test.
stop  EZ.006: All dlinks/parameters tried but failed; resetting to def values.
  EZ.009: *** Restarting Router ***

No Protocols Configured. Entering Quick Config

Router Quick Configuration for the following:
o   Interfaces
o   Bridging
        Spanning Tree Bridge (STB)
        Source Routing Bridge (SRB)
        Source Routing/Transparent Bridge (SR/TB)
        Source Routing Transparent Bridge (SRT)
o   Protocols
        IP (including OSPF, RIP and SNMP)
o   Booting

Event Logging will be enabled for all configured subsystems
with logging level 'Standard'

*********************************************************
Interface Configuration
*********************************************************

Type 'Yes' to Configure Interfaces
Type 'No' to skip Interface Configuration
Type 'Quit' to exit Quick Config Configure Interfaces? (Yes, No, Quit):
[Yes] q

Quick Config Done

Config (only)>
```

Talk

Use the **talk** command to connect to other router processes, such as GWCON, MONITR, or CONFIG. After connecting to a new process, you can send specific commands to and receive output from that process. You cannot talk to the TASKER or OPCON process. See “Command History for GWCON and CONFIG Command Line” on page 1-22 for information on how to

To obtain the PID, use the OPCON **status** command. Once you are connected to the second-level process, such as CONFIG, use the intercept character, **Ctrl** **P**, to return to the * prompt.

Syntax: talk *pid*

Example: **`talk 5`**

When using third-level processes, such as IP Config or IP, use the **exit** command to return to the second level.

Telnet

Use the **telnet** command when you want to remotely attach to another router or to a remote host (*ip address*). The only optional parameter is the terminal type that you want to emulate.

A router has a maximum of five Telnet sessions: two servers (inbound to the router), and three clients (outbound from the router).

Note: To use Telnet in a pure bridging environment, you must enable Host Services.

Syntax: telnet *ip address terminal type*

Example: **`telnet 128.185.10.30`** or **`telnet 128.185.10.30  23`** or **`telnet 128.185.10.30  vt100`**

```
Trying 128.185.10.30  ...
Connected to 128.185.10.30
Escape character is '^]'
```

If you are Telneting to a non-existent IP address, the router displays:

```
Trying 128.185.10.30  ...
```

To enter the Telnet command mode, type the escape character-sequence, which is **Ctrl-]**, at any prompt.

```
telnet>
```

If you telnet into a router,

- Press ← **Backspace** to delete the last character typed on the command line.

  Note: If you use a VT100t terminal, do not press ← **Backspace** because it inserts invisible characters. You must press **Delete** to delete the last character.

- Press **Ctrl** **U** at the `telnet>` prompt to delete the whole command line entry so that you can re-enter a command.

The Telnet command mode consists of the following subcommands:

close	Close current connection
display	Display operating parameters
mode	Try to enter line-by-line or character-at-a-time mode
open	Connect to a site
quit	Exit Telnet
send	Transmit special characters ('send ?' for more)
set	Set operating parameters ('set ?' for more)
status	Print status information
toggle	Toggle operating parameters ('toggle ?' for more)
z	Suspend Telnet
?	Print help information

The **status** and **send** subcommands have one of two responses depending on whether or not the user is connected to another host. For example:

Connected to a host:

```
telnet>    status
```

Connected to 128.185.10.30 Operating in character-at-a-time mode. Escape character is ∧].

```
telnet>    send ayt
```

Note: The send command currently supports only ayt.

Not connected to a host:

```
telnet>    status
Need to be connected first.
telnet>    send ayt
Need to be connected first.
```

Use the **close** subcommand to close a connection to a remote host and terminate the Telnet session. Use the **quit** subcommand to exit the **telnet** command mode, close a connection, and terminate a Telnet session.

```
telnet>    close
```

or

```
telnet>    quit

logout
*
```

Chapter 3. The CONFIG Process and Commands

This chapter describes the CONFIG process and includes the following sections:

What is CONFIG?

The Configuration process (CONFIG) is a second-level process of the router user interface. Using CONFIG commands, you can:

- Set or change various configuration parameters
- Add or delete an interface to the hardware configuration
- Enter the Boot CONFIG command mode
- Enter the Quick Configuration mode
- Clear, list, or update configuration information
- Enable or disable console login and modem control
- Communicate with third-level processes, including protocol environments

Note: Refer to the chapter entitled "Migrating to a New Code Level" in the Maintenance Guide for information about migrating to a new code level.

CONFIG lets you display or change the configuration information stored in the router's nonvolatile configuration memory. Changes to system and protocol parameters do not take effect until you restart the router or reload the router software. (For more information, refer to the OPCON **restart** and **reload** commands in Chapter 2, "The OPCON Process and Commands" on page 2-1).

CONFIG fits into the router software structure as shown in Figure 3-1 on page 3-2.

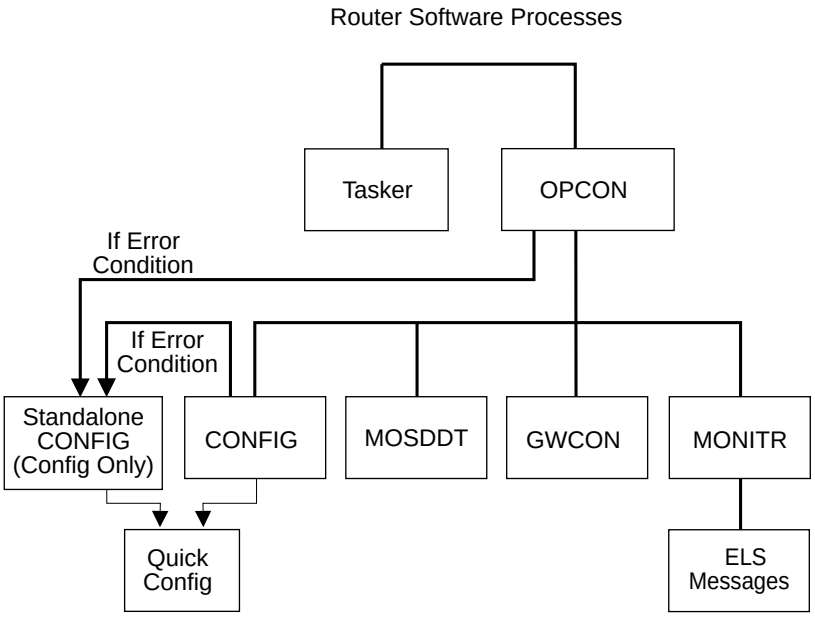


Figure 3-1. CONFIG in the Router Software Structure

The CONFIG command interface is made up of levels that are called modes. Each mode has its own prompt. For example, the prompt for the TCP/IP protocol is `IP config>`.

If you want to know the process and mode you are communicating with, press **Return** to display the prompt. Some commands in this chapter, such as the **network** and **protocol** commands, allow you to access and exit the various levels in CONFIG. See Table 3-2 on page 3-11 for a list of the commands you can issue from the CONFIG process.

Using EasyStart

EasyStart mode automatically downloads the configuration of the router from a BOOTP server. During the process the router displays the `EasyStart>` prompt and ELS messages which track the process.

1. The Network Administrator sets up the BOOTP server with records for downloading configurations. The Network Administrator must configure the BOOTP server with a valid configuration file for your type of router. For more information about configuring a BOOTP server, see “BOOTP Using a Console Terminal” on page 5-2.
2. Turn on the router and it loads itself from the IBD or the network using BOOTP.

   As soon as the operating software starts running, EasyStart begins to work if the router has no devices or protocols configured, as it would for a new router. On startup, devices are entered into the configuration automatically with default parameters.

Note: EasyStart begins when default devices are configured but no protocols are configured.

There is no manual entry into EasyStart but you can cause the router to go into EasyStart by typing the following commands at the `Config` prompt:

```
Config>clear all
You are about to clear all non Device configuration information.
Are you sure you want to do this (Yes or [No]): yes
non Device configuration cleared

Config>clear device
You are about to clear all Device configuration information
Are you sure you want to do this (Yes or [No]): yes
Device configuration cleared

*restart
Are you sure you want to restart the gateway? (Yes or [No]):  yes

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control Entering EasyStart operation.
Type 'stop' to terminate.
ELS messages are automatically displayed in this mode.

EasyStart>

  EZ.001: Starting.
  EZ.007: Waiting up to 30 seconds for devices to pass self-test.

stop
  EZ.009: *** Restarting Router ***


No Protocols Configured. Entering Quick Config

Router Quick Configuration for the following:
o   Interfaces
o   Bridging
        Spanning Tree Bridge (STB)
```

If you are in EasyStart and you enter **stop**, the router restarts and puts you into Quick Config automatically. For more information about Quick Config, see “Qconfig” on page 3-32.

If you are in EasyStart and you enter **pause**, the router suspends the EasyStart process. Enter **restart** to resume the process. Only suspend EasyStart for debugging purposes.

Config-Only Mode

Config-Only mode is a way to back out of a bad configuration that is causing the router to crash during start-up. Use the Config-Only mode ***only*** to change devices or data links (that is, for unsupported devices) or to reduce memory use (for *no memory* crashes) such as routing table sizes, packet sizes, and receive buffer allocations.

Note: Config-Only is provided only for getting a subset of configuration commands when a config problem causes the router to panic, check, fail, or detect a bug. Do ***not*** use Config-Only mode for general router configuration; many of the device-related commands are disabled in Config-Only mode and some may cause a crash.

Automatic Entry Into Config-Only Mode

Figure 1-3 on page 1-11 illustrates how the router enters Stand-alone Configuration (Config-Only mode), EasyStart, or Quick Configuration.

Config-Only mode is entered when the router detects a problem during operation or during router initialization.

Any of the following situations will cause the router to enter into Config-Only mode:

- The software load does not match the device configuration. More particularly, an attempt is made to configure a device or data link that is unsupported by the software load.
- Devices are configured but there are no protocols configured.
- Deletion of all router interface information.

If the router entered into the configuration-only mode because an unsupported device has been configured:

- Change the device information to match the hardware installed in (and supported by) the router, or change the unsupported device to “null device.”
- Enter the **Restart** command from the `Config (only)>` prompt.
- The router will automatically enter into OPCON (*).

If no protocols or devices are configured, except for default devices, the router comes up in EasyStart. For additional information, see “Using EasyStart” on page 3-2.

Manual Entry Into Config-Only Mode

To enter the Config-Only mode, take any one of the following actions:

- Reload the router with no configuration.
- Reload the router with no interfaces configured.
- Reload the router with no protocols configured.

Note: If autoboot is enabled and if you press **Ctrl** **C** while the software is loading, you go directly to the bootstrap monitor > prompt without seeing the text and you can skip step 1. Otherwise, the following text appears:

```
PROM Load/Dump Program * Revision: 1.15 *
Copyright IBM Corp. 1994, 1997
Host **VL-51*  loading

Using Ethernet at ( 81600, 94).
Trying host 128.185.210.125, via 128.185.123.28
       file loads/latest-gen.rbx2-multisna.ldc
.loading
.....................................................
....
```

1. If boot information is missing, the software will load from the IBD. If the first IBD file is invalid, such as a config file, the software will go to the manual load prompt:

```
No valid boot records found, attempting IBD load
Loading using IBD Load Image "vl2-15.cfg"
Bad record header 0

No valid server configured -- Entering manual mode
Device types available:

         IBD
         Token Ring
         WAN

Device type:
```

2. Press **Ctrl** **C** to go to the bootstrap monitor. The > prompt appears.

```
Bootstrap Monitor v1.15
Copyright IBM Corp. 1994, 1997
>
```

3. Boot to Config-Only mode.

```
>bc

PROM Load/Dump Program * Revision: 1.15 *
Copyright IBM Corp. 1994, 1997
Host **VL-51*  loading

Device types available:

        IBD
        Ethernet
        WAN

Device type [Ethernet]:
Connector Type (AUI/RJ45) [AUTO_CONFIG]:
Interface IP address [128.185.123.51]: 10.1.155.22
IP mask [FFFFFF00]:
Boot from host [128.185.210.125]:
Via gateway [128.185.123.28]: 43
Boot file name [loads/latest-gen.rbx2-multisna.ldc]:

Using Ethernet at (     0, 0).
Trying host 128.185.210.125, via 128.185.123.28
       file loads/latest-gen.rbx2-multisna.ldc
.loading
.............................................................
Starting at 1040010

The Standalone Configuration Process. You are here because
The watchdog timer timed out and/or Autoboot not selected

Config (only)>
```

See Chapter 5, “Boot Options” on page 5-1 for more detail.

During initial start-up, if no devices are configured the router comes up in Config-Only mode. If no protocols are configured the router comes up in Config-Only mode and automatically enters Quick Config. Quick Configuration is explained in the next section.

Quick Configuration

Quick Configuration (Quick Config) provides a minimal set of commands that allow you to configure various devices (interfaces), bridging protocols, routing protocols, and booting records present in the router load. It also allows configuration of some of the interfaces, booting information, and if the corresponding hardware feature is installed, Console Modem-Control.You can also configure an SNMP community

with WRITE_READ_TRAP access. This is useful during initial setup because the configuration program uses SNMP SET commands to transfer the configuration.

Table 3-1 lists what Quick Config supports.

Table 3-1. Quick Config Capabilities

Devices	ATM Protocols	Bridging Protocols	Routing Protocols	Booting	Dial Circuits
Token-Ring, Ethernet, PPP, FR, Multilink-PPP	LAN Emulation	STB, SRT, SRB, SR/TB	IP, IPX, DNA IV	TFTP, BootP,	FR, PPP, Dial-Out, Dial-In

The Quick Config complements the existing configuration process by offering a shortcut. This shortcut allows you to configure the minimum number of parameters for these devices, bridging protocols, and routing protocols and booting records without having to exit and enter the different configuration processes. The other parameters are set to selected defaults.

Situations that call for the router to be quickly configured are:

- Blank or corrupted configuration memory, such as when one of the following situations occurs:
  - The router is configured for the first time.
  - Voltage fluctuations resulted in corruption of configuration memory.
  - The CPU board, which contains the configuration memory chip, was replaced in the router.
- Demonstration purposes, for which the router needs to be quickly configured to demonstrate its capabilities.
- Bench-marking tests to get the tests going without having to learn the router's operating system commands.

Quick Config operates as follows:

- It asks a series of questions with default values.
- It offers a short-cut to the detailed configuration of the normal mode command set.

Quick Config sets a number of default parameters based upon how you answer the configuration questions. What cannot be configured with Quick Config can be configured using Config after exiting it.

You cannot delete Quick Config information from within Quick Config. However, you can correct information either by exiting and returning to Quick Config, or by entering the **restart** command as a response to some Quick Config questions.

For complete information on using the Quick Config software, see Appendix A, "Quick Configuration Reference" on page A-1.

There are two ways to get into Quick Config: automatically from EasyStart or manually.

Automatic Entry Into Quick Config Mode

If you are in EasyStart and you type **stop**, the router enters Quick Config automatically. See Figure 1-3 on page 1-11.

What you cannot configure with Quick Config you can configure using CONFIG processes after exiting Quick Config.

You cannot delete Quick Config information; but you can correct it by exiting and returning to Quick Config.

Manual Entry Into the Quick Config Mode

You might want to get to Quick Config manually to demonstrate the router's capabilities, reconfiguring on the fly to benchmark tests without having to learn the router's operating system commands.

To enter Quick Config, type **qconfig** at the `Config>` prompt.

Exiting from Quick Config Mode

To exit Quick Config, restart by entering **r** from any prompt. Follow the queries until you enter **no** and then enter **q** to quit. The router returns to either the `Config (only)>` or the `Config>` prompt.

Configuring User Access

The router configuration process allows for a maximum of 50 user names, passwords, and levels of permission. Each user needs to be assigned a password and level of permission. There are three levels of permission: *Administration, Operation,* and *Monitoring*.

For more information, see the **add user** command on page 3-16.

Technical Support Access

If you are the system administrator, when you add a new user for the first time, you are asked if you want to add Technical Support access. If you answer yes, Technical Support is granted the same access privileges that you have as system administrator.

The password for this account is automatically selected by the software and is known by your service representative. This password can be changed using the **change user** command; however, if you do change the password, customer service cannot provide remote support. For additional information on the use of the **change user** command, see "Change" on page 3-18.

Configuring Spare Interfaces

Occasionally, you may need to configure a new interface along with its bridging and routing protocols without having to restart the device. You can accomplish this by configuring a number of ***spare interfaces*** on your device. Spare interfaces are useful when:

- You are adding dial circuits to your device.

Use spare interfaces to add new V.25bis or ISDN dial circuits on an existing V.25bis or ISDN interface.

- You are adding ATM LAN Emulation clients.

  Use spare interfaces to add Token-Ring or Ethernet ATM LAN Emulation clients to an existing ATM interface.

To configure a spare interface:

1. Access the CONFIG process by entering **talk 6**.
2. Configure the number of spare interfaces for the device using the **set spare-interfaces** command.
3. Exit the CONFIG process by pressing **Ctrl-P**.
4. Restart the device.

Example:

```
* talk 6
Config> set spare 2
Config>
*restart
Are you sure you want to restart the gateway? (Yes or [No]) yes
```

When the device restarts, the spare interfaces are installed as null devices.

To use one of the spare interfaces:

1. Access the CONFIG process by entering **talk 6**.
2. Add a dial circuit using the **add device** command.
3. Configure the spare interface by using the **net** command to configure the interface or add ATM LAN Emulation clients.
4. Configure the various protocols and features using the **protocol** and **feature** commands.
5. Exit the CONFIG process by pressing **Ctrl-P**.
6. Access the GWCON process by entering **talk 5**.
7. Bring the new interface online to the network using the **activate** command.

The following example shows how to configure and activate a new dial circuit on which the IP protocol is enabled. The dial circuit and IP protocol configuration are not shown.

Example:

```
*talk 6
Config> add device dial-circuit
Config> net 6
Circuit configuration
Circuit config>

⋮
 Here you would configure the dial circuit

Circuit config> exit
Config> protocol ip
IP>


⋮
 Here you would configure the IP protocol on the dial circuit.

⋮
IP> exit
Config>
*talk 5
+activate 6
```

Restrictions for Spare Interfaces

The activate command cannot be used to bring a new interface online to the network under the following circumstances:

- You have already entered a **delete interface** command. The device must be restarted if ***any*** interface has been deleted. You cannot delete a spare interface (indicated by ***null*** in list displays).
- The spare interface is the only interface that enables a protocol or feature. The protocol or feature must already be enabled on an existing interface before it can be used by a spare interface.
- The new spare interface has a header size or trailer size greater than the sizes for other interfaces.
- There is not enough memory to allocate receive buffers for the new interface.

In these cases, you must restart the device to bring the new interface online.

You can configure the following interfaces as spare interfaces, but you cannot bring them online to the network using the **activate** command:

- Dial-out
- Multilink PPP

You must restart the device to bring these interfaces online.

You can configure the following protocols on spare interfaces, but you cannot bring them online to the network using the **activate** command:

- BGP
- OSI/DECnet V
- SDLC Relay
- XTP

There are also limitations on certain functions. These limitations are:

APPN and DECnet IV	To activate these protocols on a spare interface, you must first activate the interface and then configure the protocol on the activated interface.

Bandwidth Reservation (BRS)	To configure BRS on a spare interface, you must enable BRS on each network interface where Frame Relay circuits will be active before activating the spare interface. After activating the spare interface, you can then use BRS configuration commands to make changes like adding a traffic class or assigning a protocol to a traffic class.
Frame Relay	• You cannot activate an FR dial circuit interface unless the dial circuit's base net is already active. • An activate for an FR dial circuit will fail if the frame size, MAC header, or trailer required by the spare interface is larger than other dial circuits already assigned to the base net. • If data compression is not already active in the device, data compression will not work on a spare interface defined for data compression.
IPX	When you configure IPX on a spare interface, the device will have to be restarted in the following cases: • IPX is disabled globally when you activate this interface. • IPX is disabled on all other interfaces when you activate this interface. • You configure static routes or static services on this interface. • You configure interface filters on this interface.
PPP	• You cannot activate a multilink PPP interface. • If data compression is not already active in the device, data compression will not work on a spare interface defined for data compression. • You cannot activate a spare PPP interface if the device's global buffer is too small to support a 1500-byte PPP MRU. • You cannot activate a PPP dial circuit interface unless the dial circuit's base net is already active. • An activate for a PPP dial circuit will fail if the frame size, MAC header, or trailer required by the spare interface is larger than other dial circuits already assigned to the base net.
Bridging	• Bridging was not already active. • NetBIOS filters are defined on the spare interface. • LNM was enabled on the spare interface. • The spare interface caused a change to the bridge personality or behavior (for example, adding SR port to pure TB bridge or SR-TB conversion enabled).

Entering and Exiting CONFIG

To enter CONFIG from OPCON (*):

1. At the OPCON prompt, enter the **status** command to find the PID of CONFIG. (See page 1-5 for a sample output of the **status** command.)

   ```
   * status
   ```

2. Enter the OPCON **talk** command and the PID for CONFIG:

   ```
   * talk 6
   ```

The console displays the CONFIG prompt (`Config>`). Now, you can enter CONFIG commands. If the prompt does not appear, press the **Return** key again. To exit

CONFIG and return to the OPCON prompt (*), enter the intercept character. (The default is **Ctrl** **P**.)

Entering the Desired Protocol Configuration Process

For information on accessing a particular protocol's configuration process, see "Protocol" on page 3-31.

CONFIG Commands

This section describes each of the CONFIG commands. Each command includes a description, syntax requirements, and an example. The CONFIG commands are summarized in Table 3-2.

After accessing the CONFIG environment, enter the configuration commands at the `Config>` prompt.

Table 3-2. CONFIG Command Summary

Command	Function
? (Help)	Lists the CONFIG commands or lists the options associated with specific commands.
Add	Adds an interface to the router configuration, or a user to the router.
Boot	Enters Boot CONFIG command mode.
Change	Changes a user's password or a user's parameter values associated with this interface. Also changes a slot/port of an interface.
Clear	Clears configuration information.
Delete	Deletes an interface from the router configuration or deletes a configured user.
Disable	Disables login from a remote console,
Enable	Enables login from a remote console,
Environment	Monitors the operational temperature of the router if it has two service ports.
Event	Enters the Event Logging System configuration environment.
Feature	Provides access to configuration commands for independent router features outside the usual protocol and network interface configuration processes.
List	Displays system parameters, hardware configuration, a complete user list (including PPP users).
Network	Enters the configuration environment of the specified network.
Patch	Modifies the router's global configuration.
Protocol	Enters the command environment of the specified protocol.
Qconfig	Initiates the Quick Config process.
Set	Sets system-wide parameters for buffers, host name, inactivity timer, packet size, prompt level, number of spare interfaces, baudrate, logging disposition and level, restart count,location, and contact person.
Time	Keeps track of system time and displays it on the console.
Unpatch	Restores patch variables to default values.
Update	Updates the current version of the configuration.

? (Help)

Use the **?** (**help**) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
ADD
BOOT
CHANGE
CLEAR
DELETE
DISABLE
ENABLE
EVENT
FEATURE
 LIST
NETWORK
PATCH
PROTOCOL
QCONFIG
SET
TIME OF DAY PARAMS
UNPATCH
UPDATE
WRITE
```

Example: **list ?**

```
devices
configuration
patches
users
v25-bis-address
```

Add

Use the **add** command to add an interface to the configuration, or user-access. This command also recreates device records if the configuration is inadvertently lost.

Syntax: add
- device . . .
- isdn-address . . .
- ppp-user
- user . . .
- v25-bis-address
- v34-address

device *device_type*

The **add device** command is used to create virtual interfaces like dial circuit interfaces. You must enter the interface device type (device_type) and you may be prompted for additional configuration parameters. See "Configuring the Network Interface" on page 1-19 for information about configuration parameters and supported device types.

If you enter **add device ?**, a list of supported device types is displayed.

All device and protocol configuration information related to network interfaces is stored by interface number. Any changes made to interface numbers will invalidate much of the device configuration information in the protocols.

```
Config> add device dial-circuit
Adding device as interface 8
Defaulting Data-link protocol to PPP
Use "set data-link" command to change the data-link protocol
Use "net 8" command to configure circuit parameters
```

isdn-address *address-name network-dial-address network-subdial-address*

Adds the local and remote numbers of the ISDN end-points that will be communicating with your router.

address-name

Can be anything (such as a description of the port).

network-dial-address

The telephone number of the local or the destination port.

network-subdial-address

The additional part of the telephone number, such as an extension, that gets interpreted when the interface connects to a PBX; this parameter is optional.

Note: You can use punctuation, such as parentheses and dashes, but the punctuation is not significant (the router uses only the numbers).

```
Example: add isdn-address line 1 local
Assign network dial address [0 - 32 digits]? 1 2345 67
Assign network subdial address [0 - 19 digits]? 98765
```

ppp_user

Adds a user profile to the local PPP user data base. You need to configure PPP users if you are using PPP authentication protocols, PPP encryption or the Dial-In Access to LANs (DIALs) feature and want the PPP user data base to be locally stored and managed by the device. If you want PPP user information to be obtained from a RADIUS, TACACS, or TACACS+ server then you should configure the Authentication feature instead of configuring local PPP users.

A user profile stored locally on the device consists of the following:

User Name

Name to identify user.

Password

A password known to the user and the device. The password can be up to 32 characters in length and consist of any alphanumeric character. The password is case sensitive.

Type of Route

Either “Host Route” or “Net Route.”

A host route is generally applied for single-user access. A net route is generally applied to a network access. A net route allows you to enter a net mask.

User IP Address

IP address to be assigned to a user.

A user profile-based IP address to offer to a dial-in client if requested. There are a number of ways for a 2210 to obtain an IP address for a

dial-in client. See “IP Control Protocol” on page 33-13 for more information.

Net-Route Mask
Mask for a network user.

If the dial-in user is connecting to a DIALs-enabled PPP interface, the router automatically adds a temporary static route to that client for the duration of the PPP session. Typically, this static route has a net mask of 255.255.255.255, which implies that there is a single IP host at the other end of the PPP link. However, the net mask can be overridden. If configured, this mask is used when adding the temporary route. An example of is a small router with a single network of hosts that dials into a DIALs-enabled router. The single route to the small office router will be automatically installed based on the user profile, making it unnecessary to configure routing protocols between the two hosts and cutting down on routing traffic overhead over a potentially slow link.

Hostname
Hostname to be sent to the Proxy DHCP server for use by Dynamic DNS. See Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1 for more information.

Time-Allotted
The length of time a DIALs user can be connected. This is the total for this session, and should not be confused with an inactivity timer.

Callback type
Call back method, either “Roaming” or “Required.”

Dial-Out
enable dial-out.

This parameter is specific to clients using the DIALs dial-out client. Enabling dial-out for a ppp-user allows this user to access a modem-pool of dial-out circuits. See Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1 for more information.

Encryption
enable encryption.

You add a PPP user for each remote router or DIALs client that can connect to the device you are configuring.

You are prompted for the PPP user name, password, IP address, and encryption key if encryption should be enabled for the user.

When the DIALs feature is in the software load, you are asked if this is a DIALs user.

- If you are adding a user for a DIALs client then you are prompted for the hostname, type of route, network mask, connect time, call-back information, and dial-out capability.
- If you are not adding a user for a DIALs client then the type of route, netroute mask, hostname, time allotted, callback and dial-out capability do not apply and the user profile is created with these functions disabled.

See Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1 for more information.

The input parameters are used as follows:

- The PPP user name and password are used during PPP authentication. See “PPP Authentication Protocols” on page 33-7.
- The encryption key is used by the PPP Encryption Control Protocol (ECP). See “Overview of Encryption” on page 33-14.
- The IP address is the address to be assigned to the user.

  A user profile-based IP address to offer to a dial-in client if requested. There are a number of ways for a 2210 to obtain an IP address for a dial-in client. See “IP Control Protocol” on page 33-13 for more information.
- The net mask is entered when a dial-in user is a network type. The mask defaults to 255.255.255.255 for a single user.

  If the dial-in user is connecting to a DIALs-enabled PPP interface, the router automatically adds a temporary static route to that client for the duration of the PPP session. Typically, this static route has a net mask of 255.255.255.255, which implies that there is a single IP host at the other end of the PPP link. However, the net mask can be overridden. If configured, this mask is used when adding the temporary route. An example of is a small router with a single network of hosts that dials into a DIALs-enabled router. The single route to the small office router will be automatically installed based on the user profile, making it unnecessary to configure routing protocols between the two hosts and cutting down on routing traffic overhead over a potentially slow link.
- The hostname to be sent to the Proxy DHCP server for use by Dynamic DNS. See Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1 for more information.
- The time allotted is used to restrict the amount of time that a PPP user can stay connected.
- The call back parameters are used to specify whether the router will call back the user and what number to call back. See “Configuring PPP Callback” on page 33-10 for additional information.

You can add up to 500 PPP users.

Example: Adding a PPP user when the DIALs feature is not in the software load

```
Config> add ppp_user
Enter user name:  []? sam
Password:
Enter password again:
IP address:  [0.0.0.0]? 192.9.200.44
Enable encryption for this user/port (y/n) [No]:y
Encryption key should be 16 characters long.
Encryption Key (16 characters ) in Hex(0-9, a-f, A-F):
Encryption Key again (16 characters) in Hex(0-9, a-f, A-F)

PPP User Name: sam
   IP address: 192.9.200.44
   Encryption: Enabled
```

Example: Adding a PPP user when the DIALs feature is in the software load

```
Config>add ppp_user
Enter user name:  []? robert
Password:
Enter password again:
Is this a DIALs user (Yes, No): [y]?
Type of route (hostroute, netroute) [hostroute]
IP address: [0.0.0.0]? 192.9.200.44
Enter hostname for dynamic DNS:  []? robert
Give 'robert' default time allotted ? (Yes, No): [Yes] no
Number of minutes online allotted (0=unlimited):   [0]? 5
Enable Callback for 'robert' ? (Yes, No): [No] yes
Type of Callback (Roaming Callback, Required Callback) [Roaming Callback]
Will 'robert' be able to dial-out ? (Yes, No): [No] yes
Enable encryption for this user/port (y/n) [No]:y
Encryption key should be 16 characters long.
Encryption Key (16 characters ) in Hex(0-9, a-f, A-F):
Encryption Key again (16 characters) in Hex(0-9, a-f, A-F):

    PPP User Name: robert
  User IP Address: 192.9.200.44
   Net-Route Mask: 255.255.255.255
         Hostname: robert
     Time-Alotted: 5 minutes
   Call-Back Type: Roaming Callback
         Dial-Out: Enabled
       Encryption: Enabled

Is information correct? (Yes, No, Quit): [No] yes

User 'robert' has been added
```

user *user_name*

Gives a user access to the router. You can authorize up to 50 users to access the router. Each user_name is eight characters and is case-sensitive.

When the first user is added, console login is automatically enabled. Each user added must be assigned one of the permission levels defined in Table 3-3.

Table 3-3. Access Permission

Permission Level	Description
Administrator (A)	Displays configuration and user information, adds/modifies/deletes configuration and user information. The Administrator can access any router function.
Operator (O)	Views router configuration, views statistics, runs potentially disruptive tests, dynamically changes router operation, and restarts the router. Operators cannot modify the permanent router configuration. All actions can be undone with a system restart.
Monitor (M)	Views router configuration and statistics but cannot modify or disrupt the operation of the router.
Tech Support	Allows your service representative to gain access to the router if a password is forgotten. Cannot be assigned to users.

Note: To add a user, you must have administrative permission. You do not have to reinitialize the router after adding a user.

Example:
```
add user John
Enter password:
Enter password again:
Enter permission (A)dmin, (O)perations, (M)onitor [A]?
Do you want to add Technical Support access? (Yes or [No]):
```

Enter password
Specifies the access password for the user. Limited to 80 alphanumeric characters and is case-sensitive.

Enter password again
Confirms the access password for the user.

Enter permission
Specifies the permission level for the user: A, O, or M (see Table 3-3 on page 3-16).

Do you want to add Technical Support access?
This is only an option if the user has a Dial In Access load. See Table 3-3 on page 3-16.

v25-bis-address
Adds the local and remote numbers of the V.25bis end-points that will be communicating with the router. The network *address-name* can be anything, such as a description of the port. You can use any string of up to 23 printable ASCII characters. The *network-dial-address* is the telephone number of the local or destination port. For more information, see Chapter 43, “Using and Configuring the V.25bis Network Interface” on page 43-1.

Note: You can use punctuation, such as parentheses and dashes, but the punctuation is not significant (the router uses only the numbers).

```
Example: add v25-bis-address
 remote-site baltimore 1-909-555-0983
```

v34-address
Adds the local and remote numbers of the V34 end-points that will be communicating with the router. The network *address-name* can be anything, such as a description of the port. You can use any string of up to 23 printable ASCII characters. The *network-dial-address* is the telephone number of the local or destination port. You can enter up to 31 characters that are in the valid dial characters for the connected modem. For more information, see Chapter 45, “Using and Configuring the V.34 Network Interface” on page 45-1.

Note: You can use punctuation, such as parentheses and dashes, but the punctuation is not significant (the router uses only the numbers).

Example: add v34-address

```
Assign address name [1-23] chars []? remote-site-baltimore
Assign network dial address [1-20 digits][]? 1-909-555-1234
```

Boot

Use the **boot** command to enter the Boot CONFIG command environment. For Boot CONFIG information, see Chapter 4, “The Boot CONFIG Process and Commands” on page 4-1.

Syntax: boot

Example:
```
boot
TFTP Boot/dump configuration
Boot config>
```

Change

Use the **change** command to modify an interface in the configuration, change your own password, or change user information.

Syntax: change device . . .

password

ppp_user . . .

device dial-circuit

Allows you to change a device interface into a *NULL* interface (an interface for which the configuration information is ignored) or to change a NULL interface, that was originally a dial circuit interface, back to a dial circuit interface.

```
Example: change device dial-circuit
Interface number [0]? 3
Defaulting Data-link protocol to PPP
```

```
Example: change device null
Interface number [0]? 1
```

password

Modifies the password of the user who is now logged in.

Note: To change a user password, you must have administrative permission.

```
Example: change password
Enter current password:
Enter new password:
Enter new password again:
```

Enter current password

Specifies your current password.

Enter new password

Specifies your new password.

Enter new password again

Specifies your new password again for confirmation. If your confirmation does not match the previous new password, the old password remains in effect.

ppp_user

Changes the information for a specific PPP user.

Syntax: change ppp_user encryption-key
parameters
password

encryption-key

Changes the encryption key for a PPP user. The following example shows the dialog for changing an encryption key.

```
Example: Change Encryption key
Config>
Config>change ppp_user encryption-key
Enter user name:  []? leslie
Enable encryption for this user/port (y/n) [No]:y
Encryption key should be 16 characters long.
Encryption Key (16 characters ) in Hex(0-9, a-f, A-F):
Encryption Key again (16 characters) in Hex(0-9, a-f, A-F):
User 'leslie' has been updated
Config>
```

parameters

Changes all of the ppp-user options for a user. This parameter works similar to the **add ppp_user** except that the values shown within the **[]** are the current values and the change command does not verify the changes or list them back to you when you are done. See “Add” on page 3-12 for details about the **add ppp_user** command.

password

Changes the password for the PPP user.

```
Example: Change password
Config>
Config>change ppp_user password
Enter user name:  []? sam
Password:
Enter password again:
User 'sam' has been updated
Config>
```

user

Modifies the user information that was previously configured with the **add user** command.

Note: To change a user, you must have administrative permission.

```
Example: change user
User name: []
Change password? (Yes or No)
Change permission? (Yes or [No])
```

Clear

Use the **clear** command to delete the router’s configuration information from nonvolatile configuration memory.

Attention: Use this command only after calling your service representative.

To clear a process from nonvolatile configuration memory, enter the **clear** command and the process name. To clear all information from configuration memory, except for device information, use the **clear all** command. If you want to clear all information, including the device information, use the **clear all** command and then the **clear device** command.

The **clear user** command clears all user information except the router console login information. This is left as enabled (if it was configured as enabled) even though the default value is “disabled.”

Notes:

1. To clear user information, you must have administrative permission.
2. There may be other items in the list, depending upon what is included in the software load.

Syntax: clear

- all
- ap2 (AppleTalk 2)
- arp (ARP)
- asrt (Adaptive Source Route Protocol)
- appn (Advanced Peer-to-Peer Networking)
- atm (Asynchronous Transfer Mode)
- auth (Authentication)
- bgp (Border Gateway Protocol)
- boot
- brs (Bandwidth Reservation)
- cmprs (Data Compression)
- dls (Data Link Switching)
- device
- dialer-circuit
- dn (DECnet)
- dvmrp (Distance Vector Multicast Routing Protocol)
- els (Event Logging System Information)
- environment
- fr (Frame Relay)
- hdlc
- ip (IP)
- ipx (Novell IPX)
- isdn
- osi (OSI)
- ospf (OSPF routing protocol)
- ppp (Point-to-Point)
- sdlc
- snmp
- srb (Source Route Bridge)
- srly (SDLC Relay)
- stb (Spanning Tree Bridge)
- tcp/ip-host
- time (Time of day information)
- user
- v25bis
- v34
- vines (Banyan VINES)
- wrs (WAN Restoral feature)
- x25
- xtp

Example: `clear els`

```
You are about to clear all Event Logging configuration information
Are you sure you want to do this (Yes or No):
```

Note: The previous message appears for any parameter configuration you are deleting.

Delete

Use the **delete** command to remove an interface from the list of devices stored in the configuration, or to remove a user. To use the **delete** command, you must have administrative permission.

Syntax: delete
- interface . . .
- isdn-address
- ppp_user . . .
- user . . .
- v25-bis-address
- v34-address

interface *intfc#*

To delete an interface, enter the interface or network number as part of the command. (Dial circuit is the only device type that can be deleted.) To obtain the interface number that the router assigns, use the **list device** command.

The delete interface command deletes the device configuration and any protocol information for that interface. However, the router will continue to run the previous configuration until it is reloaded.

isdn-address *address-name*

Removes a previously added ISDN address.

```
Example: delete isdn-address remote-site-XYZ
```

Note: If the *address-name* contains spaces (for example, **remote site XYZ**), you cannot enter the command on one line. Type `delete isdn-address` and press **Return**. Then enter the name when prompted.

ppp_user *user_name*

Deletes a user from the PPP user data base.

```
Example: delete ppp_user haag
Config> delete ppp_user haag

user [haag] deleted

Config>
```

user *user_name*

Removes user access to the router for the specified user.

```
Example: delete user mary
Are you sure you want to delete user 'mary' ?(Y / N)
```

v25-bis-address *address-name*

Removes a previously added V25bis address.

```
Example: delete v25-bis-address remote-site-baltimore
```

Note: If the *address-name* contains spaces (for example, **remote site Baltimore**), you cannot enter the command on one line. Type `delete v25-bis-address` and press **Return**. Then enter the name when prompted.

v34-address *address-name*

Removes a previously added V34 address.

```
Example: delete v34-address remote-site-newyork
```

Note: If the *address-name* contains spaces (for example, **remote site New York**), you cannot enter the command on one line. Type `delete v34-address` and press **Return**. Then enter the name when prompted.

Disable

Use the **disable** command to prevent being prompted for a login from a remote console and to disable modem control . The **disable** command also disables a specified interface. If the router has two service ports and you use the **disable modem-control** command, specify either **service1** or **service2**.

You can also use the disable command to disable an interface.

Syntax: disable console-login
interface . . .
modem-control

console-login

Disables the user from being prompted for a user ID and password on the physical console. The default is disabled.

```
Example: disable console-login
```

interface *interface#*

Causes the specified interface to be disabled after issuing the **restart** command. The default is enabled.

```
Example: disable interface 2
```

modem-control *service1* or *service2*

Disables monitoring of modem control lines on the console port. The default is disable.

```
Example: disable modem-control
```

If the router has two service ports, specify to which service port you connected the modem, either **service1** or **service2**. To disable *both* service ports, disable them separately.

```
Example: disable modem-control service1
```

Enable

Use the **enable** command to allow login from a remote console, enable modem control, and enable a specified interface.

Specify **enable modem-control carrier-wait** or **enable modem-control ring-wait**. For routers with two service ports, also specify **service1** or **service2**.

Syntax: enable console-login
interface . . .
modem-control

console-login

Enables the user to be prompted for a user ID and password on the physical console. This is useful for security situations. If you do not configure any administrative users and you enable this feature, the following message appears:

```
Warning: Console login is disabled until an
administrative user is added.
```

By disabling the console login, a lock-out situation is prevented.

```
Example: enable console-login
```

interface *interface#*

Causes the interface to be enabled after issuing the **restart** command. command.

```
Example: enable interface 2
```

modem-control carrier-wait OR ring-wait *service1* OR *service2*

Sets up the router for login on the physical console, if the physical console is connected to the router through a modem. Before using this command, be sure to:

- Set your modem for auto-answer.
- Verify that the console baud rate is equal to the modem baud rate.
- Verify that the cable connecting the modem to the router is configured correctly.
- Turn echo off by using the ATE0 command.
- Run in quiet mode by using the ATQ1 command.
- Verify that any necessary jumpers are set. Refer to your router's *User's Guide* more information.

The router automatically hangs up the modem when you log out. Also, if your modem becomes disconnected from the router while you are using it, the router logs you out.

Specify the service port for both the **enable modem-control carrier-wait** and the **enable modem-control ring-wait** commands. For routers with two service ports, also specify to which service port you connected the modem, either **service1** or **service2**. To enable *both* service ports, enable them separately.

Note: No console connection can be made with the router after enabling modem control unless you clear all configuration and restart the router.

You can tell the router to wait for the carrier-detect signal from the modem before sending Request to Send. This is the standard method of modem control.

```
Example: enable modem-control carrier-wait service1
```

You can tell the router to wait for the ring-indication signal before raising Request to Send or Data Terminal Ready. This is provided for countries requiring an earlier handshake.

```
Example: enable modem-control ring-wait
```

```
Example: enable modem-control ring-wait service2
```

Environment

Note: This command is to be invoked ***only*** for routers with two service ports.

The Environment System lets you monitor the operational temperature of the router. You can configure high and low temperature thresholds; when the operational temperature of the router exceeds one of these thresholds, the router emits periodic ELS events until the operational temperature of the router falls below (for high temperature conditions) or rises above (for low temperature conditions) the threshold.

Under extremely warm conditions, a chip holds the router in a reset state which prevents it from operating. To ensure correct operation of the router, a temperature chip allows it to operate in the range –55°C to +85°C (–67°F to +185°F). However, only the upper limit affects the operation of the router; a temperature chip shuts off the router at 85°C or above and the router does not come back on until it is at 80°C or below. Although extreme cold does not interrupt the router's operation, –55°C is the lowest temperature the chip registers.

The **environment** command displays the `ENV config>` prompt which has three available commands, **list**, **set**, and **exit**. Enter the **exit** command to return to the `Config>` prompt.

Syntax: environment

Example:
```
environment
Environment System user configuration
ENV config>
```

Use the **set** command to set the high and low temperatures at which the system raises an alarm condition.

Note: The reset temperature level is factory set. You cannot modify it.

There are three **set** commands, **set-low-temp-threshold**, **set high-temp-threshold**, and **set recalc-temp-interval**. Enter thresholds in degrees Celsius (°C). Use the **set recalc-temp-interval** command to set the amount of time between successive temperature readings.

Example: `set high-temp-threshold`

```
Enter the High Temperature Threshold.
Range: <-34/c..85C> [72]? 80
```

Example: `set low-temp-threshold`

```
Enter the Low Temperature Threshold.
Range: <-55/c..74C> [-40]? 0
```

Example: `set recalc-temp-interval`

```
Enter the Time in seconds between successive temperature
recalculations.
Range: <10..86400> [60]? 30
```

Temperature ranges vary depending on the environment in which you place the router. Use the **environment** command described on page 3-23 to determine your router's natural operating range over time.

Use the **set high-temp-threshold** command to receive ELS messages before the router resets. The value should be about 10°C less than the maximum (85°C) so that you get some ELS messages before the router resets itself.

Use the **set low-temp-threshold** command to receive ELS messages. The value should be about 10°C more than the minimum (–55°C) so that you get some ELS messages. The router does not reset itself on cold temperatures.

Use the **list** command to display the settings.

Example: `list`

```
Current Ambient Temperature: 53C (127F)

Recalculate temperature interval:  30 seconds (approx)

High Temperature Alarm Threshold:   80C (176F)
Low Temperature Alarm Threshold:     0C (32F)
 (Hysteresis value: +/- 5C)
```

Hysteresis is the amount the temperature must change past the set alert threshold before the alert condition is cleared. For a device with two service ports, hysteresis value is fixed at ± 5 degrees. For example, if you specify a high-temp-threshold of 75°C, you will get ELS messages from 75 degrees and above. The temperature must go below 70 degrees before the condition is cleared (75 - 5 = 70). If you specify a low-temp-threshold of -10°C, you will get ELS messages from -10 degrees and below. The temperature must move above -5 degrees before you no longer get ELS messages (-10 +5 = -5).

Event

Use the **event** command to enter the Event Logging System (ELS) environment so that you can define the messages that will appear on the console. Refer to Chapter 8, “Using and Configuring the Event Logging System (ELS)” on page 8-1 for information about ELS.

Syntax: event

Example:

```
event
Event Logging System user configuration
ELS config>
```

Feature

Use the **feature** command to access configuration commands for specific router features outside of the protocol and network interface configuration processes.

All 2210 features have commands that are executed by:

- Accessing the configuration process to initially configure and enable the feature, as well as perform later configuration changes.
- Accessing the console process to monitor information about each feature, or make temporary configuration changes.

The procedure for accessing these processes is the same for all features. The following information describes the procedure.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release.

Example:

```
feature ?
WRS
BRS
MCF
CMPRS
DIALS
AUTH
Feature name or number [1] ?
```

To access a feature's configuration prompt, enter the **feature** command followed by the feature number or short name. Table 3-4 on page 3-26 lists available feature numbers and names.

Table 3-4. IBM 2210 Feature Numbers and Names

Feature Number	Feature Short Name	Accesses the following feature configuration process
0	WRS	WAN Restoral/Reroute
1	BRS	Bandwidth Reservation
7	CMPRS	Data Compression
9	DIALS	Dial-In-Access to LANs
10	AUTH	Authentication

Example:
```
feature mcf
MAC filtering user configuration
Filter Config>
```

Once you access the configuration prompt for a feature, you can begin entering specific configuration commands for the feature. To return to the CONFIG prompt, enter the **exit** command at the feature's configuration prompt.

Syntax: feature *feature# OR feature-short-name*

Example:
```
feature 2
  or
feature MCF
```

List

Use the **list** command to display configuration information for all network interfaces, or configuration information for the router.

Syntax: list
- devices
- configuration
- isdn-address
- patches
- ppp_users
- users
- v25-bis-address
- v34-address

devices

Displays the relationship between an interface number and the hardware interface. You can also use this command to check that a device was added correctly issuing the **add** command.

Example: `list devices`

```
Ifc 0 Ethernet                  CSR  81600, CSR2  80C00, vector 94
Ifc 1 WAN X.25                  CSR  81620, CSR2  80D00, vector 93
Ifc 2 WAN X.25                  CSR  81640, CSR2  80E00, vector 92
Ifc 3 WAN PPP                   CSR 381620, CSR2 380D00, vector 125
Ifc 4 WAN Frame Relay           CSR 381640, CSR2 380E00, vector 124
Ifc 5 Token Ring                CSR 600000, vector 95
```

Note: The number of receive buffers noted are exceptions from the receive buffer defaults. The **set receive buffers** command is discussed under "Set" on page 3-32.

configuration

Displays configuration information about the router.

Example: `list configuration`

```
Hostname: acctg
Maximum packet size: [autoconfigured]
Maximum number of global buffers: [autoconfigured]
Number of spare interfaces: 0
Number of Restarts before a Reload/Dump: 64
Logging disposition: detached
Console baudrate:  9600 (Autobaud)
Console inactivity timer (minutes): 0
Physical console login: disabled
Modem Control Enabled, using CARRIER-WAIT type control
Contact person for this node: [none]
Location of this node: [none]

Configurable Protocols:
Num Name  Protocol
0   IP    DOD-IP
3   ARP   Address Resolution
4   DN    DNA Phase IV
6   VIN   Banyan VINES
7   IPX   NetWare IPX
8   OSI   ISO CLNP/ESIS/ISIS
9   DVM   Distance Vector Multicast Routing Protocol
10  BGP   Border Gateway Protocol
11  SNMP  Simple Network Management Protocol
12  OSPF  Open SPF-Based Routing Protocol
20  SDLC  SDLC/HDLC-Relay
22  AP2   AppleTalk Phase 2
23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge
24  HST   TCP/IP Host Services
25  LNM   Lan Network Manager
26  DLS   Data Link Switching
27  XTP   X.25 Transport Protocol
28  APPN  Advanced Peer-to-Peer Networking [HPR]
29  NHRP  Next Hop Routing Protocol
30  APPN  Advanced Peer-to-Peer Networking [ISR]

Configurable Features:
Num Name  Feature
0   WRS   WAN Restoral
1   BRS   Bandwidth Reservation
2   MCF   MAC Filtering
6   QOS   Quality of Service
7   CMPRS Data Compression Subsystem
8   NDR   Network Dispatching Router
10  AUTH  Authentication

27616 bytes of configuration memory free
```

isdn-address

Displays the current ISDN address configurations.

```
Example: list isdn-address
Address assigned name     Network Address     Network Subdial Address
---------------------     ---------------     -----------------------
remote site XYZ            1 2345 67           98765
```

patches

Displays the values of patch variables that have been entered using the **patch** command.

```
 Example: list patches
Patched variable            Value

ping-size                   60
ping-ttl                    59
ip-default-ttl              60
ethernet-security           3
rip-static-suppress         3
```

ppp_users

Lists specific PPP user profile parameters.

Example: List of PPP users when DIALs is not in the software load

```
Config> list ppp_users
List (Name, Verb, User, Addr, Encr):

       PPP User Name: joe
     User IP Address: Interface Default
          Encryption: Not Enabled
```

Example: List of PPP users when DIALS is in the software load

```
Config> list ppp_users
List (Name, Verb, User, Addr, Call, Time, Dial, Encr):

       PPP User Name: joe
     User IP Address: Interface Default
      Net-Route Mask: 255.255.255.255
            Hostname: <undefined>
        Time-Alloted: Box Default
      Call-Back Type: Not Enabled
            Dial-Out: Not Enabled
          Encryption: Not Enabled
```

When you enter **list ppp_users**, the software will prompt you to enter one of the following:

Name
List all of the names in the database.

Verb
List verbose information about each user. List all information pertaining to each user profile.

User
List verbose information about a single user.

Addr (address)
List IP address information for each user, including IP Address, net mask and hostname.

Call (callback)
List callback information for each user, including the type of callback and number.

Time
List time allowed configured for each user.

Dial (dialback)
List dial out status for each user.

Encr (encryption)
List whether encryption is enabled for each user.

users

Displays the users configured to access the system.

```
Example:
list users
USER           PERMISSION
joe            operations
mary           administrative
peter          monitor
```

v25-bis-address

Displays the current V25bis address configurations. The V25bis address configuration consists of the network address and network address name for a local port (serial line interface) or destination port. The network address is the telephone number of the local or destination port. The network address name can be anything, such as the description of the port. For more information, see Chapter 43, “Using and Configuring the V.25bis Network Interface” on page 43-1.

```
Example:
list v25-bis-address
Address assigned name             Network Address
----------------------            ---------------
v25-1                             8982800
v25-2                             8980001
westboro                          1-666-555-4444
```

v34-address

Displays the current V34 address configurations. For more information, see Chapter 45, “Using and Configuring the V.34 Network Interface” on page 45-1.

```
Example:
list v34-address
Local Network Address Name    = v403
Local Network Address         = 1-508-898-2403
```

Network

Use the **network** command to enter the network interface configuration environment for supported networks. Enter the interface or network number as part of the command. (To obtain the interface number, use the CONFIG **list device** command.) The appropriate configuration prompt (for example, `TKR Config>`) will be displayed. See the network interface configuration chapters in this book for complete information on configuring your types of network interfaces.

Note: Whenever you change a user-configurable parameter, you must **restart**

Syntax: network *interface#*

Example:
```
network 0
TKR config>
```

Note: Not all network interfaces are user-configurable. For interfaces that you cannot configure, you receive the message: `That network is not configurable.`

Patch

Use the **patch** command for modifying the router's global configuration. Patch variables are recorded in nonvolatile configuration memory and take effect immediately; you do not have to wait for the next restart of the router. This command should be used only for handling uncommon configurations. Anything that you commonly configure should still be handled by using the specific configuration commands. The following is a list of the current patch variables documented and supported for this release.

Syntax: patch

Example:
```
patch ping-size
New value [0]?
```

bgp-subnets new value

If you want the BGP speaker to advertise subnet routes to its neighbors, set *new value* to 1. The default is 0.

New value The new value for the variable that you are patching.

dls-ignore-lfs

When set to 1, DLSw ignores the "largest frame" size bits in source-routed frames when setting up a circuit. This avoids circuit setup problems with some older LAN products that do not set these bits correctly. The default is 0.

ethernet-security

When set to a non-zero value, zeros the padding that is applied to Ethernet packets whose data portion is less than the physical minimum of 60 bytes. This may be required for security reasons. Default: 0.

ip-default-ttl

The TTL used in packets that are originated by the router. The default is 64.

Note: It is preferable to set this parameter with the **set ttl** IP configuration command. (See the "Set" section of the "Using and Configuring IP" chapter of *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1*.) This patch variable remains for compatibility with configurations from older releases.

ip-mtu

This parameter limits the IP MTU size to the specified value. When this parameter is set, the IP MTU size on a given network interface is set to the lesser of the ip-mtu value and the largest value that network interface's configured frame size can accommodate.

more-lines

The number of lines to display on the console when listing the IP routing table, which uses a "more pipe" (¦).

mosheap-lowmark

This parameter specifies the percentage of free MOS heap memory, at which the device notifies the operator that an out-of-memory error is imminent. This notification allows the operator to take action to free up MOS heap memory before the device receives an error and stops.

When the operator receives notification, the operator can reconfigure the router and then reboot, minimizing the outage to the network. Specifying 0 for this parameter suppresses this warning.

Valid Values: 0 to 100

Default Value: 10

ospf-import-rate
Number of routes imported per second.

ping-size
The size of the data portion (that is, excluding IP and ICMP headers) of the ICMP PING packet that is sent via the IP>**ping** command. Default: 56 bytes. (The size of the PING data can also be entered as a parameter of the **ping** command as described in the "Ping" section of the "Monitoring IP" chapter of *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1*.)

ping-ttl
The TTL (time-to-live) sent in PINGs by the IP>**ping** command. Default: 64. (The TTL can also be entered as a parameter of the **ping** command as described in the "Ping" section of the "Monitoring IP" chapter of *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1*.

ppp-echo
When set to 1, the device will not send PPP Echo Requests on any PPP interface. PPP Echo Requests are sent to remote devices as part of PPP maintenance to ensure the remote device is operational. Consider enabling this variable when running PPP on a slow line and using that line to transmit large data packets such that the PPP maintenance packets are not exchanged often enough to keep the PPP interface up.

relax-jate
Relaxes JATE ISDN restriction.

rip-static-suppress
When set to a non-zero value, static routes will not be advertised by RIP over a given interface unless the IP config> **enable send static** command is given for the interface. This changes the semantics of the **enable send static** command. When rip-static-suppress is equal to 0 (the default), the list of the routes advertised via RIP is the union of those specified by the interface's RIP flags.

Note: You must specify the complete name of the patch variable that you want to change. You cannot use an abbreviated syntax for the patch name.

Protocol

Use the **protocol** command at the Config> prompt to enter the configuration environment for the protocol software installed in the router. The **protocol** command followed by the desired protocol number *or* short name lets you enter a protocol's command environment. After you enter this command, the prompt of the specified protocol appears. From the prompt, you can enter commands specific to that protocol. Table 1-2 on page 1-16 lists examples of protocol numbers and names. To return to Config>, enter the **exit** command.

Notes:

1. To see the names and numbers of the protocols in your software load, at the `Config>` prompt, enter **list configuration**.
2. When you change a user-configurable parameter, you must restart the router for the change to take effect. To do so, enter the **restart** command at the OPCON prompt (*).

   The changes you make through CONFIG are kept in a configuration database in nonvolatile memory and are recalled when you restart the router.

Syntax: protocol *prot#*

```
Example:
 protocol 7
or
protocol ipx
IPX config>
```

Qconfig

Use the **qconfig** command to initiate Quick Config. Quick Config allows you to configure parameters for interfaces, boot records, andbridging and routing protocols without entering separate configuration environments.

Syntax: qconfig

Note: For complete information on using the Quick Config software provided with your router, refer to the *IBM 2210 Nways Multiprotocol Router Installation and Initial Configuration Guide*

Set

Use the **set** command to configure various system-wide parameters.

Syntax: set
- baudrate
- contact-person . . .
- data-link . . .
- down-notify . . .
- global-buffers
- hostname
- inactivity-timer
- input-low-water
- location . . .
- logging disposition
- logging level
- packet-size
- prompt-level
- receive-buffers
- restart-count
- spare-interfaces

baudrate

Sets the console baud rate. The valid options are 0 (for autobaud), 300, 1200, 2400, 4800, 9600, 19200, and 38400.

```
Example: set baudrate 2400
```

contact-person *sysContact*

Sets the name or identification of the contact person for this managed SNMP node. There is a limit of 80 characters for the *sysContact* name length.

This variable is for information purposes only and has no effect on router operation. It is useful for SNMP management identification of the system.

```
Example: set contact-person nautilus
```

data-link *type interface#*

Select the data link type for a serial interface. The type can be one of:

- FRAME-RELAY
- PPP
- SDLC
- SRLY
- V25BIS
- V34
- X25

Interface# is the number of the interface you are configuring.

```
Example: set data-link PPP 3
```

down-notify *interface# # of seconds*

Allows the user to specify the number of seconds before declaring an interface as being down. The normal maintenance packet interval is 3 seconds, and it takes four maintenance failures to declare the interface as down.

The **set down-notify** command is used primarily when tunneling LLC traffic over an IP network using OSPF. If an interface goes down, OSPF cannot detect it fast enough because of the length of time that it takes for an interface to be declared down. Therefore, LLC sessions would begin to timeout. You can set the down-notify timer to a lower value, allowing OSPF to sense that an interface is down quicker. This enables an alternate route to be chosen more quickly, which will prevent the LLC sessions from timing out.

Note: If the **set down-notify** command is executed on one end of a serial link, the same command must be performed at the other end of the link or the link may not come up and stay up.

Interface#

The number of the interface you are configuring.

# of seconds

The down notification time value that specifies the maximum time that will elapse before a down interface is marked as such. Large values will cause the router to ignore transient connection problems, and smaller values will cause the router to react more quickly. The range of values is 1 to 300 seconds and the default is 0, which sets the 3-second period. Setting the down notification time to 0 will restore the default time for that interface.

The **list devices** command will show the down notification time setting for any interface that has the default value overridden.

```
Example: set down-notify 4 3
```

global-buffers *max#*

Sets the maximum number of global packet buffers, which are the packet buffers used for locally originated packets. The default is to autoconfigure for the maximum number of buffers (up to 1000). To restore the default, set the value to 0. To display the setting for global-buffers, use the **list configuration** command.

```
Example: set global-buffers 30
```

hostname

Adds or changes the router name. The router name is for identification only; it does not affect any router addresses. The name must be :

- Less than 78 characters and is case sensitive
- Set before storing the router's configuration memory in IBD.

```
Example: set hostname sales
```

inactivity-timer *# of min*

Changes the setting of the Inactivity Timer. The Inactivity Timer logs out a user if the remote or physical console is inactive for the period of time specified in this command. This command affects only consoles that require login. The default setting of 0 turns the inactivity timer off, indicating that no logoff is performed, no matter how long a console remains inactive.

```
Example: set inactivity-timer 3
```

input-low-water *interface # low # of receive buffers*

Allows you to configure the value of the low number of receive buffers, or packets, on a per-interface basis, thus overriding the default values.

The memory allocation strategy changes to conserve buffers when the number of free buffers is equal to or less than the low or low-water mark value. When a packet is received, and the current value of the interface is less than the low water value, then that packet is eligible for flow control (dropping).

The range of values is 1 to 255. The default is both platform and device specific. Setting the value to 0 restores the autoconfigured default.

Interface # is the number of the interface you are configuring. *Low # of receive buffers* is the low water value.

Lowering the value will make it less likely that packets from this interface will be dropped when sent on congested networks. However, lowering the value may negatively affect performance if it drops packets to the extent that the receive queue is frequently empty. Raising the value has the opposite effect.

Type the **QUEUE** or **BUFFER** command at the GWCON prompt (**+**) to show the low setting.

```
Example: set input-low-water 4 7
```

location *sysLocation*

Sets the physical location of an SNMP node. There is a limit of 80 characters for the *sysLocation* name length. This variable is for information purposes only and has no effect on router operation. It is useful for SNMP management identification of the system.

```
Example: set location atlanta
```

logging disposition *setting*

Changes the SRAM record for the default logging disposition. This command affects the MONITR process (that is, it changes the default setting at startup).

The logging disposition settings are as follows:

- *console* writes to the console (equivalent to the OPCON **divert 2 0** command).
- *detached* holds the data and does not print it (equivalent to the OPCON **halt 2** command).
- *flush* discards the data (equivalent to the OPCON **flush 2** command).

If you have a printing terminal attached to the router's console port, you can obtain a hard copy of the startup messages by setting the logging disposition to **console**, and restarting the router.

```
Example: set logging disposition console
```

logging level *#*

Controls the output of messages that have not yet been converted to the ELS. (Refer to Chapter 8, "Using and Configuring the Event Logging System (ELS)" on page 8-1 for more information about the ELS.) The logging level is recorded in the configuration. When the router is powered on or restarted, the logging level takes effect and determines message output. The default logging level is 76. Logging level 0 equates to no logging level.

```
Example: set logging level 76
```

packet-size *max packet size in bytes*

Establishes or changes the maximum size for global buffers and receive buffers. If you specify a value of 0 as the maximum packet size, the size of receive buffers for an interface is based on that interface's configured packet size and the packet size of global buffers are autoconfigured. If you specify a non-zero value, the configured value is used as the global buffer packet size and any interfaces that have a configured packet size that is larger than the maximum packet size will use the maximum packet size for their receive buffers. A value of 0 (for autoconfigure) is the default.

Attention: Use this command only under direct instructions from your service representative. ***Never*** use it to reduce packet size – ***only*** to increase it.

```
Example:
set packet-size
What is the maximum packet size (in bytes) [0]?
Packet size updated successfully
```

prompt-level *user-defined-name*

Adds a user-defined name as a prefix to all operator prompts, replacing the hostname.

The user-defined-name can be any combination of characters, numbers, and spaces up to 80 characters. Special characters may be used to request additional functions as described in Table 3-5 on page 3-36.

```
Example:
set prompt
What is the new MOS prompt [y]? AnyHost 99
AnyHost 99 Config>
```

Table 3-5. Additional Functions Provided by the Set Prompt Level Command

Special Characters	Function Provided by the Set Prompt Level Command

$n	Displays the hostname. This is useful when you want the hostname included in the prompt. For example:
	`Config> **set prompt**`
	`What is the new MOS prompt [y]? **$n**`
	`hostname:: Config>`

$t	Displays the time. For example:
	`Config> **set prompt**.`
	`What is the new MOS prompt [y]? **$t**`
	`02:51:08[GMT-300] Config>`

$d	Displays the current date-month-year. For example:
	`Config> **set prompt**.`
	`What is the new MOS prompt [y]? **$d**`
	`26-Feb-1997 Config>`

$v	Displays the software VPD information in the following format: `program-product-number Feature xxxx Vx Rx.x PTFx RPQx`
$e	Erases one character *after* this combination within the user-defined prompt.
$h	Erases one character *before* this combination within the user-defined prompt.
$_	Adds a carriage return to the user-defined prompt.
$$	Displays the $.

Note: You can combine these commands. For example:
`Config> **set prompt**.`
`What is the new MOS prompt [y]? **$n::$d**`
`hostname::26-Feb-1997 Config>`	

receive-buffers *interface # max #*

Adjusts the number of private receive buffers for most interfaces. The range is 5 to 255.

Note: This command is not applicable for ISDN Primary Rate Interfaces. For ISDN PRI, the interface handler determines the value based on the number of dial circuits configured.

(On some devices, the maximum value is restricted further, as shown in 3-6.) To restore the default, set the value to 0. The **set receive-buffers** command can be used to increase the receive performance of an interface. In addition, this command can be used to reduce flow control drops when the router is forwarding many packets from a fast interface to a slow interface. The effect of this command is visible on the GWCON **buffer** command.

Attention: Use this command only under direct instructions from your service representative.

```
Example: set receive-buffers 4 30
```

Table 3-6. Default and Maximum Settings for Interfaces

Interface	Default	Maximum
ATM	80	80
ETH	50	50
Serial	24	24
TKR	40	120

restart-count

Establishes the number of times a router will restart due to a serious error before dumping (if enabled) and reloading. In general, the restart-count should not be changed. The default is 64.

```
Example: set restart-count
```

spare-interfaces ***n***

Defines ***n***, the number of spare interfaces, for this device. See "Configuring Spare Interfaces" on page 3-7 for additional information.

Example: `set spare 5`

Time

Use the **time** command to set the 2210 system clock and date, and to display the values on the user console. These values can then be used to time-stamp ELS messages.

Note: The 2210 has a hardware clock that maintains the date and time after router reinitialization.

```
Syntax:  time
host . . .
list
offset
set . . .
sync . . .
```

host *IP_address*

Sets the IP address of the RFC 868-compliant host that will be used as the time source. This is the address of a host which will respond to an empty datagram on UDP port 37 with a datagram containing the current time.

```
Example: time host 131.210.1.4
```

list

Displays all configured time-related parameters. This includes the current time (if set) and the source of the time (operator or IP address from which time was last received).

```
Example: time list
05:20:27  Wednesday December 7, 1994
Set by: operator
Time Host:  131.210.4.1
Sync Interval: 10 seconds GMT
Offset:  -300 minutes
```

offset *minutes*

Defines the time zone, in minutes, offset from GMT (Greenwich Mean Time). Note that values west of GMT are negative. For example, EST is 5 hours earlier than GMT, so the command would be **time offset -300**.

```
Example: time offset
minutes from GMT (-720 to 720) [0]? -300
```

set <*year month date hour minute second*>

Prompts you to set the current time. If you do not specify the entire time in the command, you are prompted for the remaining values. You can change the date as shown in the following example.

```
Example: time set
year [1996] 1997
month [12]?
date [6]? 7
hour [11]? 12
minute [3]?
second [2]?
```

sync *seconds*

Sets the period, in seconds, at which the router will poll the time host for the current time.

```
Example: time sync 300
```

Unpatch

Use the **unpatch** command to restore the values of the patch variables entered with the **patch** command to their default values. See the **patch** command in "Patch" on page 3-30.

Note: You ***must*** specify the long name of the patch variable to be restored.

Syntax: unpatch *variable_name*

Example: **`unpatch ethernet-security`**

Update

Use the **update** command to update the configuration memory when you receive a new software load. Follow the instructions on the release notice sent with the software. The **update** command is the last command that you enter when loading new software. After you enter this command, the console displays a message indicating configuration memory is being updated.

Syntax: update version-of-SRAM

Example: **`update version-of-SRAM`**

```
Updating configuration memory to V15.2 [X104]
```

Chapter 4. The Boot CONFIG Process and Commands

This chapter describes the Boot CONFIG process. This chapter includes the following sections:

What is Boot CONFIG?

Router nonvolatile configuration database memory contains the data that controls the router boot and dump capabilities. The Boot CONFIG commands allow you to modify this data.

Using Boot CONFIG commands, you can:

- Add, modify, or remove entries from the boot and dump configuration database.
- Disable or enable network memory dumping and assign a unique name to the dump files.
- Use the TFTP protocol to transfer (using the **TFTP** command or **copy** command) configuration information between router memory and remote hosts.
- View the current boot and dump configuration database.
- Store file images to the Integrated Boot Device (IBD).
- Store the current image to the IBD.
- Leave the Boot CONFIG command environment and return to the CONFIG process.
- List the contents of the IBD.
- Delete files from the IBD.
- Copy files to and from the local router memory and another local router memory or host file system.
- Save any changes you have made to system and protocol parameters.

Changes made to system and protocol parameters through Boot CONFIG take effect when you restart the router or when you reload the router software.

Configuring Booting

Boot files are the same as load image files. A boot file contains the software load for the router and resides on a host server, or an IBD. The host server is, for example, any PC, router, or workstation, that is running the IP protocol and TFTP. The boot configuration database can contain an entry for each boot file, configured

using the **add** command. Each entry contains the address of the host server, the next hop router, and the timeout, path, and filenames of the boot files.

You can configure more than one boot file in the boot configuration database by specifying the path and name of each boot file (using the **add** command described on page “Add” on page 4-12). If you have more than one host server, you can use a different host server to boot the router when another host server cannot be reached over the network.

To configure booting:

1. Add an address record, using the **add address** command from the `Boot config>` prompt, that specifies the interface from which you want it to boot.
2. Add the boot record, using the **add boot-entry** command from the `Boot config>` prompt, specifying the host address, next hop router (if necessary), and the path and filename of the host.

Using a Device as a Boot Server

A device can also function as a boot server. Devices that do not have an IBD can obtain their load files or boot files from a router that has an IBD. Use the **add boot-entry** command to designate the location of the router with the boot file. Make sure that you include the entire path name of the load file with this command. On a router with the load in IBD, this is `IBD/`*`filename`*`.`

How the BOOTP Forwarding Process Works

BOOTP (documented in RFC 951) is a bootstrap protocol used by a router or a diskless workstation to learn its IP address, the location of its boot file, and the boot server name. A device can act as a *BOOTP client* or as a *BOOTP relay agent* for another device. The following sections describe these two processes.

A Device as a BOOTP Client

A device acts as a BOOTP Client when it needs to find the location of the boot file and boot server. You can specifically configure the device’s boot PROM configuration record so the router can act as a BOOTP Client, or it can become a BOOTP Client if, during booting, it does not contain a valid file name and path to the location of the boot file and server. When either of these two conditions exists, the router broadcasts a UDP packet over one of its LAN interfaces to the *BOOTP server* that contains the path name of the boot file and server.

The following describes the BOOT client forwarding process:

1. The BOOTP client copies its MAC address (either Ethernet or Token Ring) into a BOOTP packet (UDP packet) and broadcasts it onto the local LAN. BOOTP is running on top of UDP.
2. The BOOTP server receives the request and looks up the client's Ethernet address in its database. If found, it formats a BOOTP reply containing the client's IP address, the location of its boot file, and the boot server name. The reply is then sent back over the LAN to the BOOTP client.

   Note: If multiple hops are required before reaching the BOOTP server, a BOOTP relay agent receives the packet. BOOTP relay agent is explained in the next section.

3. When the router receives the BOOTP reply packet, it uses the information it contains to initiate a TFTP request to the boot server.

A Device as a BOOTP Relay Agent

If BOOTP request requires multiple hops before reaching the BOOTP server, the BOOTP relay agent routes the packet via IP to all BOOTP servers that it knows about. If any other router receives this packet while it is being routed via IP, it will examine the packet to determine whether it is a BOOTP packet and route that packet toward the BOOTP servers that it knows about. The following describes the BOOTP relay agent forwarding process:

1. A device acting as the local BOOTP relay agent, receives the BOOTP request packet from the BOOTP client, modifies the checksum, places an IP header on the packet with the relay agent's IP address copied into the body of the BOOTP request, and routes the packet to all BOOTP servers.
2. The BOOTP servers receive the request and look up the client's MAC address in their database. If a server finds the client's address, it formats a BOOTP reply containing the client's IP address, the location of its boot file, and the boot server name. The reply is then sent to the BOOTP relay agent.
3. The BOOTP relay agent receives the reply, makes an entry in its ARP table for the client, and then forwards the reply to the BOOTP client.
4. The client then continues to boot using the information that is contained in the BOOTP reply packet to initiate a TFTP request to the boot server.

Enabling/Disabling BOOTP Forwarding

To enable or disable BOOTP forwarding on the router, enter the following appropriate command at the IP configuration prompt:

```
IP Config> enable bootp
IP Config> disable bootp
```

When enabling BOOTP, you are prompted for the following values:

- Maximum number of application hops you want the BOOTP request to go.

  This is the maximum number of BOOTP relay agents that can forward the packet. This is ***not*** the maximum number of IP hops to the BOOTP server. A typical value for this parameter is 4.
- Number of seconds you want the client to retry before you forward the BOOTP request. *This parameter is not commonly used.* A typical value for this parameter is 0.

After accepting a BOOTP request, the router forwards the BOOTP request to each BOOTP server. If there are multiple servers configured for BOOTP, the transmitting server replicates the packet.

Configuring a BOOTP Server

The BOOTP server is either an AIX or UNIX host with a *bootpd* daemon, or a DOS host (running software available from FTP Software). The BOOTP server contains a file (maintained by the network administrator) that lists all the BOOTP clients that this server is responsible for, and their associated IP addresses, boot file locations, and boot server names.

When the BOOTP server receives a BOOTP request, it matches the MAC address of the client with the MAC address in its BOOTP file. If a match occurs, the server constructs a BOOTP reply and adds the client's IP address, along with the location of the Boot server and boot filename. If a match does not occur, the packet is dropped.

To add a BOOTP server to the router's configuration, enter the following command at the IP configuration prompt:

```
IP Config> add BOOTP-SERVER [IP address of server]
```

You can configure multiple servers. In addition, if you know only the network number of the server, or if multiple servers reside on the same network segment, you can configure a broadcast address for the server using the **enable directed-broadcast** command at the `IP config>` prompt.

Using the Trivial File Transfer Protocol (TFTP)

TFTP is a file transfer protocol that runs over the Internet UDP protocol. This implementation provides multiple, simultaneous TFTP file transfers between a router's non-volatile configuration memory, Integrated Boot Device (IBD), and remote hosts.

TFTP allows you to:

- Store a configuration file from a router to a server
- Copy a configuration file from a server to a router
- Copy a configuration or load file to an IBD.

TFTP transfers involve a *client* node and a *server* node. The client node generates a TFTP request onto the network. The router acts as a client node by generating TFTP requests from the router console using the `Boot Config>` process **copy** command.

Note: The **tftp** command and the **copy** command have the same function but the syntax is different.

The client can transfer a copy of the configuration file stored in configuration memory, or any file stored in the IBD.

The server is any device (for example, a personal computer (PC), router, or workstation) that receives and services the TFTP requests. When the router acts as a server, transfers are transparent to the user. Use the ELS subsystem tftp message log to view the transfer in progress.

Note: A file server or router is not allowed to *copy* any file into another router's nonvolatile config memory or IBD. To write to the router, use the **copy** command at the destination's local `Boot config>` prompt.

Before using the **copy** command, note that:

- The device configuration must include the IP protocol and have at least one configured IP address. Also, the router must not be operating in CONFIG-Only mode.
- When a device's configuration memory is empty (i.e., initially installing the device, corrupted SRAM), you must set the following parameters to restore the device's configuration.
  1. Set the device's host name.
  2. Configure IP so that the device can reach each host with the archived configuration. The *Protocol Configuration and Monitoring Reference* explains the IP configuration commands.
- The source IP address for TFTP transfers is the device ID. This ID, by default, is a configured IP address for one of the device's network interfaces. To change the router ID, use the **set router ID** command at the `IP Config>` prompt.
- All TFTP data transfers are 512 bytes long. A data transfer of less than 512 bytes indicates an end to the transfer. A protocol, client, or remote host error generates an error packet which terminates the transfer.
- Download configuration files into the same type of router from which you are uploading the file.

Note: This implementation of TFTP does not allow you to *copy* to other routers.

Every TFTP transfer has a client and server UDP port number. When a client node generates an initial request to the server, an unused UDP port number on the client node is randomly selected as the client port. The server port is the UDP port number 69 (decimal). If a TFTP server is running on the server, it listens on UDP port 69. When the server receives a request from the network, a UDP port number currently unused on the server is randomly selected as the host port. The file transfers then occur on these two UDP ports.

Accessing Configuration Files From a Remote Host or Router

To access configuration files from a remote host or router:

1. At the `Boot config>` prompt, type **copy** and press **Enter**.
2. At the `source filename [CONFIG]?` prompt, specify the remote IP address and the pathname.

   This is the TFTP host or another router with the file in its IBD.
3. At the `destination filename [Config]?` prompt, press **Enter**.

   By pressing **Enter** you are accepting the default filename, CONFIG. For example:

```
Boot config>copy
source filename[CONFIG]?128.185.210.125:loads/configs/vl-28.cfg
destination filename [CONFIG]?
COPYing from "128.185.210.125:loads/configs/vl-28.cfg" to
"CONFIG"
COPY succeeded
```

Filename Definitions for IBD

Each file or *image* stored on the IBD must have a unique *loadname* associated with it. The file name for the IBD can contain the complete path name in addition to the file name.

Example 1: `test.cfg`
Example 2: `/usr/loads/test.ldc`

The following example shows how to store a file to the IBD at the `Boot config>` prompt:

Example: `copy 128.185.210.125:/usr/config/test.cfg ibd/test.cfg`

The router accepts any printable ASCII character as part of the file name definition, with two exceptions:

- The file name cannot begin with a numeric character
- The file name cannot contain a RETURN or LF (line feed) character.

The character string can accept a space, but it is recommended that you avoid using a space character, as this character is invisible. Another user who tries to enter the file name without the required space receives an error message.

Note: When using a IBM 2210 as a boot server for other routers, be sure to include the complete path name to the load file with the **add boot-entry** command on the booting router.

The following table contains the convention for filename extensions.

Table 4-1. Conventions for File Name Extensions

Type of File	Filename Extension
Configuration	.cfg
Load	.ldc

IBD Considerations When Transferring a File

When transferring a file to the IBD consider the following:

- A full load may not fit into one bank of the IBD.
- Any load that needs more than one bank for storage writes only to empty, numerically adjacent banks. For example, when storing a load too large for bank 2, the load is stored in bank 3, as long as bank 3 is empty.
- If an adjacent bank is unavailable to store a large load, a `TFTP Disk Full` message appears on the console, the load is not stored, and the IBD remains unchanged. Any portion of the load that was stored in a bank is then removed.

Validating the Configuration Load

There are two methods for validating an image before it is written into the device's configuration memory:

- In the first method, the device assigns an identifier, called a *Magic Number*, to each platform type for the image that is archived and the image that is being restored. If the numbers do not match, the transfer is aborted and the console displays the message `Bad Magic Number`.

- In the second method, the host name for the device that originally archived the image is compared to the host name for the device that is restoring the image. If the host names do not match, the transfer is aborted and the console displays the message:

  ```
  COPY error -
  Got hostname "<hostname>" - is this okay (Yes or [NO])? no
  ```

  This allows you to bring in the configuration from another device even if the hostname does not match. The configuration needs to be correct for your model device.

  When a transfer fails due to a lack of RAM space, the console displays an error message.

Loading an Image at a Specific Time

There may be occasions when you may want to load an image into a device on a specific day and time when you will be unavailable. You can configure the device to perform a timed load using the **timedload activate command**. Other commands allow you to view a device's scheduled load information or cancel a scheduled load. See "Boot CONFIG Commands" on page 4-10 for information on these commands.

Configuring Dumping

An important feature of the 2210 is the ability to dump the contents of system memory and processor's registers to another host during a system reset that results from a software crash, hardware failure, or by pressing the reset button.

To configure dumping, do the following from the `Boot config>` prompt:

1. *Add address.*

   This can be the same as the boot address used in configuring booting.

2. *Add a dump entry.*

   This is the location of the host or server that is going to receive the dump file. You can add a dump entry with the **add dump-entry** command. The average size of a dump file is 8 Mb.

3. *Enable dumping.*

   Dumping will not work unless you enable it using the **enable dumping** command. Dumping will remain enabled until you use the **disable dumping** command to terminate it.

Dump Files

Dump files contain the contents of the system memory and processor registers.

When the device crashes and dumping is enabled, the contents of memory are written to a remote host using TFTP. Each dump entry contains the location of the host server and the path, timeout, and file names for the dump files.

You can configure the device to automatically append a unique character string to the dump file names. This prevents an existing dump file from being overwritten by subsequent dumps. However, unique naming of the dump files can cause the server's disk to become full if there are successive dumps. Unique naming may

also be incompatible with the security requirements of some TFTP servers. Some servers require that a file already exist on the server to allow writing the dumps.

Dump files are for diagnostic purposes only. Enable the device's dump and unique-naming capabilities only on the advice of your Customer Service representative.

TFTP Server, Boot and Dump Directories

You must create directories on the destination server to contain the boot and dump files. These directories must reside on a host server and the boot directories must be globally readable and the dump directories globally writable. The boot and dump functions use the TFTP protocol. Your TFTP server may impose additional restrictions.

Installing Software/Code

To download a new load module from a server into the IBD, perform the following steps:

1. Install your load file into a server that is reachable by the device. Make sure the TFTP daemon is running in your server. On the device, issue the following commands at the router console:
2. At the OPCON prompt (*****):
   a. Enter **status** to display the Config process ID (PID).

   ```
   * status
   ```

   b. Enter **talk** and the Config PID to access the `Config>` command environment.

   ```
   * talk 6
   ```

3. At the `Config>` prompt, enter **boot**. This will access the `Boot config>` command environment.

   ```
   Config> boot
   Boot config>
   ```

4. At the `Boot config>` prompt, enter **add address** to specify an IP address over which the device can boot. This needs to be done only once for each interface you want to be able to use. It should not be done each time you want to get a new load module.

   You will then be prompted for the following information:

   - Interface number. This is the number of the interface over which the router will transfer the file.
   - New address. This is the IP address of this interface.
   - Net mask. This is the network mask for this interface.

   ```
   Boot config> add address
   Which interface is this address for [0]?
   New address [0.0.0.0] ?
   Net mask for this interface [255.255.255.0]?
   ```

   The next steps are needed only if you added a boot address. If your boot address is already configured, skip these steps and go to step 9 on page 4-9.

5. Press **Ctrl** **P** to return to the OPCON prompt (*****).

6. Enter **restart** at the OPCON prompt.
7. Enter **talk** and the Config PID.
8. Enter **boot** at the `Config>` prompt to return to the `Boot config>` command environment.
9. At the `Boot config>` prompt, enter **tftp get**. This initiates the file transfer of the load module.

   You will be prompted for the following information:

   - Local filename. For the local filename, enter the filename of the new load in the IBD.
   - Remote host. For the remote host, enter the IP address of the server.
   - Host filename. For the host filename, specify the entire path and filename on the host machine.

   ```
   Boot config> tftp get
   Local filename []?  ibd/newloadfile
   Remote host []?
   Host filename []?
   ```

10. Enter **list boot-entries** at the `Boot config>` prompt. This lists the load modules in your IBD.

    ```
    Boot config> list boot-entries
    ```

    Note the entry number of the load module in the IBD that you were using prior to receiving this load module.

    The boot database is where the router goes to determine where to get the load module from. You may have multiple entries in your database. The first entry is usually a load module in the IBD, and the second is usually a load module on a remote host or router.

11. To change the boot database pointer to the module you just loaded, enter **change boot** at the `Boot config>` prompt. This is what determines which load module is used the next time you reboot the router.

    ```
    Boot config> change boot
    ```

    You will then be prompted for the entry number of the previous module you were using in IBD. This is the entry number from step 10. The boot entry number will usually be "1".

    ```
    Change which entry?: 1
    ```

12. Enter the filename of the new load. This is the name that you specified at step 9 to store in the IBD. Filenames are case sensitive.

    ```
    remote host or IBD load name:
    ```

13. Enter **exit**.

    ```
    Boot config> exit
    ```

14. Press **Ctrl** **P** to return to the OPCON prompt (*****).
15. Enter **restart** to make sure the configuration change from the "change boot" command takes effect.
16. Enter **reload** to load the device with the new load module.
17. Once you are confident with the new load, you can create space in your IBD for future loads by erasing the previous load:

    a. Enter **talk 6**.

b. Enter **boot**.

```
Config> boot
```

c. Enter **list ibd** to list the content of the banks. Note the number of the banks where the previous load is stored.

```
Boot config>list ibd
```

d. Enter **erase** and either the previous load name or the bank numbers. For example, to erase from bank 36 to 50, enter:

```
Boot config> erase 36-50
```

Entering and Exiting Boot CONFIG

To enter the Boot CONFIG command environment, use the CONFIG **boot** command. When the router's software is initially loaded, it is running in the OPCON process, signified by the * prompt. From the * prompt:

1. Enter **talk 6**.
2. At the `Config>` prompt, type **boot**.
3. At the `Boot config>` prompt, type **?**. The following appears:

```
ADD
CHANGE
COPY config
DESCRIBE
DELETE
DISABLE
ENABLE
ERASE
LIST
LOAD
STORE
TIMEDLOAD
TFTP
EXIT
```

To return to the CONFIG process, type **exit**.

Boot CONFIG Commands

This section describes the Boot CONFIG commands. Each command includes a description, syntax requirements, and an example. Table 4-2 on page 4-11 summarizes the Boot CONFIG commands.

After accessing the Boot CONFIG environment, enter the boot configuration commands at the `Boot config>` prompt.

Table 4-2. Boot CONFIG Commands

Command	Function
? (Help)	Displays a list of the commands available from that prompt level.
Add	Adds a boot interface IP address to a specified interface, host boot entry, or host dump entry.
Change	Changes the boot interface IP address, network boot entry data, or network dump entry data.
Copy	Copies boot files and configuration files to or from remote routers and hosts or between resources within the router.
Describe	Displays information about the stored loadfile images in the IBD.
Delete	Deletes a network boot interface address, a host boot entry, or host dump entry.
Disable	Disables memory dump or unique naming of the dump files.
Enable	Enables memory dump or unique naming of dump files.
Erase	Erases a stored image on an IBD bank.
List	Displays all network boot addresses, all boot and dump configuration data, the contents of the IBD, BOOTP name settings, and scheduled image load information.
Load	Copies a boot file from the IBD to RAM or copies a boot file from a remote host to RAM.
Store	Copies the boot file from RAM to the IBD.
Timedload	Schedules an image load into the device on a specific day and time, cancels a scheduled load or displays scheduled load information.
TFTP	Initiates TFTP file transfers between device memory or IBD and remote hosts.
Exit	Leaves the Boot CONFIG environment and returns to the CONFIG process.

? (Help)

Use the **?** (**help**) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: ?

```
ADD
CHANGE
COPY config
DESCRIBE
DELETE
DISABLE
ENABLE
ERASE
LIST
LOAD
STORE
TIMEDLOAD
TFTP
EXIT
```

Example:add ?

```
ADDRESS
BOOT-ENTRY
BP-DEVICE
DUMP-ENTRY
```

Add

Use the **add** command to enter boot/dump parameters into the device's configuration database.

Syntax: add

- address
- boot-entry
- bp-device
- dump-entry

address

Specifies the IP address of the interface or device over which the device can boot or dump. When you enter the **add address** command, you must supply or accept the default value of the following information:

- Interface number of the network interface
- IP address
- Network mask

To obtain the interface number (Ifc#), use the CONFIG **list devices** command. Chapter 3, “The CONFIG Process and Commands” on page 3-1 describes this command.

Note: Failure to add an address results in the device being unable to boot or dump over the network.

Remember the following:

- The first address you enter corresponds to the first boot-entry entered, the second address to second boot-entry, and so on.
- Multiple boot entries can use the same IP address (interface).
- You must enter this command if you are using the **add boot-entry**, **add dump-entry** and **load remote** commands.

```
Example: add address
Which interface is this address for [0]?
New address [0.0.0.0] ? 128.185.1.2
Net mask for this interface [255.255.255.0]?
```

boot-entry

Specifies the information needed by the device to locate the TFTP host server and retrieve the boot image file. There are several ways that a device can boot:

- If the router is booting up using software stored in its IBD, then you must specify the IBD loadname as the first boot entry in the configuration. You can configure more than one boot device. Obtain the loadname using the **list ibd** command. The loadname is case-sensitive.

```
Example: add boot-entry
remote host or IBD loadname [0.0.0.0]? 128.185.30.0
via gateway (0.0.0.0 if none) [0.0.0.0]?  0.0.0.0
timeout in seconds [3]?  10
file name [ ]?  loads/Y21.ldc
```

- If the device is booting using software stored on a TFTP server, then you must specify the IP address of the remote TFTP host server. Note that the TFTP host server can be another device with an IBD.
- If the TFTP host server is on a remote network (not directly connected to the booting router), you must specify the IP address of the next hop (router) towards the host server.

Table 4-3. Add Boot Entry Parameters

remote host or IBD loadname?	IP address of the remote host or an IBD loadname. **Note:** An IBD loadname must start with a letter. Otherwise, the system interprets the string as an IP address.
via gateway?	IP address of the first hop router, if any. If the TFTP host server is on a directly connected network, answer 0.0.0.0.
timeout in seconds?	Specifies the amount of time the device will wait before retransmission takes place. The default is 3 seconds. This may need to be set to a longer time over exceptionally slow boot paths.
file name?	The complete directory path and name of the boot image file on the TFTP host server. (The complete directory path is not necessary on some machines. The default assumes the path is `tftpboot/` which is invisible to you, so if the path is `/tftpboot/loads/name`, you type **loads/name**.) • When referencing a file stored on a UNIX-based operating system use a forward slash "/" and remember that the file name is case-sensitive. If the path requires the leading forward slash (/) use a double forward slash (//): `128.185.15.1//tftpboot/loads/name`. • When referencing a file stored on a DOS disk use a backward slash "\" and remember that the file name is not case-sensitive.

Note: To view a list of the current boot configuration, enter the Boot CONFIG **list boot** command.

```
Example: list boot-entry
remote host or IBD loadname [0.0.0.0]?  10.0.0.5
via gateway (0.0.0.0 if none) [0.0.0.0]?  12.0.0.7
timeout in seconds [3]  10
file name [ ]  loads/vl.ldc
```

bp-device

Provides a BOOTP boot-up capability as follows for retrieving the device's software from a BOOTP (Boot Protocol) device.

- If the device has never been configured or is missing it's automatic boot up configuration information and the auto-boot switch is enabled, the device will automatically attempt to use BOOTP on all LAN interfaces to retrieve its boot-up information.
- During an auto-boot, the device will try to use the information provided in the boot entries to retrieve its load image file first. If the device cannot retrieve its load image file with the information in the boot entries, it will then attempt to boot up using BOOTP.
- The interfaces selected with the **add bp-device** command depend on the locations of the BOOTP servers in the network.
- You cannot use BOOTP to boot over directly connected serial interfaces.

```
Example: add bp-device
Which interface number [0]? 1
```

dump-entry

Specifies the IP address of the remote host that will receive the dump file(s). When you enter the **add dump-entry** command, you must supply the following information:

remote host?	IP address of the remote host on which the dump file will be stored, usually same as boot server
via gateway?	If host is on a remote network (not directly connected to the booting device), you must specify the IP address of the next hop (router) towards the host. If the host is on a directly connected network, answer 0.0.0.0.
timeout in seconds?	Specifies the amount of time the device will wait before retransmission takes place. The default is 3 seconds. This may need to be set to a longer time over exceptionally slow boot paths.
file name?	Base dump path and filename (may have unique suffix appended).

To view a list of the dump configurations, enter the **list dump-entries** command.

Example:

```
add dump-entry
remote host [0.0.0.0]? 128.185.162.30
via gateway (0.0.0.0 if none) [0.0.0.0]? 128.185.160.3
timeout in seconds [3]?
file name []?  c:\dump\gertrude.dmp
```

Change

Use the **change** command to modify entries in the existing address, boot-entry, and dump-entry information without deleting and re-adding the information. You can delete and re-enter information instead of using the **change** command.

Syntax: change address
boot-entry
bp-device
dump-entry

address

Changes an existing address for a boot interface or device that was previously added. When you enter the **change address** command, you must supply the following information:

- Address entry number
- Interface number of the network interface
- IP address
- Network mask

Note: The console displays some of this information, such as the address entry number, when you enter the Boot CONFIG **list** command. To obtain the interface number (Ifc#), use the CONFIG **list devices** command. (Chapter 3, “The CONFIG Process and Commands” on page 3-1 describes this command.)

Example:

```
change address
Change which entry [1]? 1
Which interface is this address for [0]? 1
New address [192.9.1.1]? 128.185.162.1
Net mask for this interface [255.255.255.0]?
```

boot-entry

Modifies the configuration about a previously added network boot file. When you enter the **change boot-entry** command, you must supply the following information:

- Boot entry number
- IP address of the remote host
- IP address of the first hop router, if any
- TFTP retransmission timer value
- Boot file name, if different from the current file name.

Note: The console displays some of this information, such as the boot entry number, when you enter the Boot CONFIG **list boot-entries** command.

Example:

```
change boot-entry
change which entry [1]?
remote host [18.123.0.16]?
via gateway (0.0.0.0 if none) [0.0.0.0]?
timeout in seconds [3]?
file name [user/lib/gw/gwimage.ldb]?
```

bp-device

Changes the interface that is the BOOTP device. To obtain the entry number for an interface, use the **list boot-entries** command.

Example:

```
change bp-device
Change which entry [1]?
Which interface is this entry for [1]?
```

Note: For more information on the BOOTP protocol and its related processes, refer to the chapters on configuring and monitoring the IP protocol in the *Protocol Configuration and Monitoring Reference*

dump-entry

Modifies the configuration about a previously added network dump file. When you enter the **change dump-entry** command, you must supply the following information:

- Dump entry number

- IP address of the remote host
- IP address of the first hop router, if any
- TFTP retransmission timer value
- Base boot file name, if different from the current file name

Note: Use the Boot CONFIG **list dump-entries** command to display this information.

Example:

```
change dump-entry
change which entry [1]? 1
remote host [18.123.0.16]?
via gateway (0.0.0.0 if none) [0.0.0.0]?
timeout in seconds [3]?
file name [user/lib/gw/gwimage.ldb]? c:\dump\debug1.dmp
```

Copy

Use the **copy** command to copy boot files and configuration files to and from remote routers and hosts. To use the **copy** command, the device must have IP configured and running on at least one interface. The device cannot be in Config-only mode.

Syntax: copy

config
ibd/*filename*
host-ip-address/*filename*

Example 1: **Copying from a Remote Router**

```
Boot config> copy
source filename [CONFIG] 128.185.110.30/ibd/Y17.ldc
destination filename  IBD/Y17.ldc
```

Source filename and *destination filename* must be one of the following:

config	Configuration memory
ibd/filename	File name on IBD. Include the complete pathname.
IP address/remote	Remote file on TFTP host.
path and filename	Include the complete pathname.

Note: When copying a file to the IBD, the file is placed in the largest set of contiguous free banks. If no banks are available the message `COPY error - TFTP Disk Full or IBD full` appears on the console.

In the example above, get the source from a remote router whose IP address is 128.185.110.30. The IBD has a filename Y17.ldc. The colon (:) is used here as the delimiter. The *destination* has a filename of Y17.cfg.

Example 2: **Copying from a Remote Host**

```
Boot config> copy
source filename [CONFIG]  128.185.110.30/router/loads/2210.02.cfg
destination filename ibd/2210.02.cfg
```

In the example above, the source has a path and filename. The destination is an IBD.

Example 3: **Copying Within a Device**

```
Boot config> copy
source filename [CONFIG] config
destination filename [CONFIG]? ibd/2210.02.cfg
```

In the example above, the source is the configuration memory. The destination is an IBD.

config

Gets the same result as if you type copy and press the **Enter** key, except that you do not get prompted for the source filename.

Example: `copy config`

```
Boot config> copy config
destination filename [IBD/VL_1-71(p28).cfg]?
```

ibd/*filename*

Copies a boot file or configuration file from an IBD. You must include the file name.

Example: `copy ibd/vl-28.cfg`

```
Boot config> copy ibd/vl-28.cfg
destination filename [CONFIG]?
```

host-ip-address/*filename*

Copies a boot file or configuration file from a remote host. You must include the file name.

Example: `copy 128.185.110.30:/loads/test.ldc`

```
Boot config> copy 128.185.110.30:/loads/test.ldc
destination filename [CONFIG]? ibd/test.ldc
```

Delete

Use the **delete** command to remove entries from the boot and dump configuration database.

Syntax: delete

- address
- boot-entry
- bp-device
- dump-entry

address

Removes an interface address entry from the boot and dump configuration database.

When you enter the **delete address** command, a prompt appears for the entry you want to delete. The address entry number is the first number that appears on each line when you enter the **list address** command at the `Boot config>` prompt.

To verify the deletion, use the **list** command.

Example:

```
delete address
Delete which entry [1]?
```

boot-entry

Removes a boot entry from the boot and dump configuration database. When you enter the **delete boot-entry** command, a prompt appears to enter the boot-entry you want to delete. The boot-entry number is the first number that appears on each line when you enter the **list boot-entries** command at the `Boot config>` prompt.

To verify the deletion, use the **list** command.

Example:

```
delete boot-entry
Delete which entry [1]? 2
```

bp-device

Removes the specified interface as a BOOTP device.

Example:

```
delete bp-device
Delete which entry [1]?
```

Note: For more information on the BootP protocol and its related processes, refer to the chapters on configuring and monitoring the IP protocol in the *Protocol Configuration and Monitoring Reference*

dump-entry

Removes a dump entry from the boot and dump configuration database. When you enter the **delete dump-entry** command, a prompt appears for the entry you want to delete. The dump entry number is the first number that appears on each line when you enter the **list dump-entries** command at the `Boot config>` prompt.

To verify the deletion, use the **list** command.

Example: **delete dump-entry**

```
Delete which entry [1]?
```

Describe

Use the **describe** command to display information about a stored image in the IBD.

Syntax: describe *loadname*

loadname

Displays the following information about the specified loadname:

- Copyright information.
- Supported protocols, features, and data-link types.
- Supported network interfaces.

Example: **describe ibd/test.ldb**

```
        Copyright Notice .....


IBM 2210 Bridging Router   V1 R2.0[Y69]    Wed Mar 8 10:24:20
1995

Software configuration: Expanded Multi-Protocol DLSw
Includes:
 Internet Protocol - IP & OSPF
 Novell - IPX
 AppleTalk Phase 2 - AP2
 Banyan VINES - VIN
 Adaptive Source Routing Transparent Bridge - ASRT
  with NETBIOS Name Caching & Filtering
 Data Link Switching - DLSw
 SDLC Relay - SRLY
 Frame Relay
 PPP
 X.25
 V.25bis
 WAN Restoral/Reroute - WRS
 Bandwidth Reservation - BRS
 MAC Filtering - MCF
```

Disable

Use the **disable** command to disable memory dumping and the unique naming of dump files.

Syntax: disable dumping
unique-naming

dumping

Prevents the device from performing a memory dump to the network when a system failure occurs. After entering this command, use the Boot CONFIG **list all** command to verify the dump settings. Dumping is suppressed until you enter the **enable dumping** command. **Disable dumping** is the default setting.

Example: **disable dumping**

unique-naming

Suppresses the automatic unique naming of dump files. After entering this command, use the Boot CONFIG **list all** command to verify the unique naming settings. Unique naming is suppressed until you enter the **enable unique-naming** command. **Disable unique-naming** is the default setting.

Example: `disable unique-naming`

Enable

Use the **enable** command to enable memory dumping and the unique naming of dump files. This command must be used to turn on the dump operation using the data supplied with the **add dump-entry** command. Dumping should only be enabled if the device is experiencing problems of unknown origin. Dump files can be very large and consume large amounts of disk space. It takes considerable time to transfer the large dump file through the Internet to the dump host.

Syntax: enable dumping
unique-naming

dumping

Enables the router to perform a memory dump to the network when a system failure occurs. After entering this command, use the Boot CONFIG **list all** command to verify the dump settings. Dumping is enabled until you enter the **disable dumping** command.

Example: `enable dumping`

unique-naming

Enables the automatic unique naming of dump files. A random suffix (hexadecimal) from 1 to 5 characters will be added to the configured dump filename when the dump occurs.

After entering this command, use the Boot CONFIG **list all** command to verify the unique naming settings. Dump files include unique names until you enter the **disable unique-naming** command.

Example: `enable unique-naming`

Erase

Use the **erase** command to erase a stored image or an IBD bank.

Syntax: erase . . .

loadname OR *bank-number*

Erases a stored image or an IBD bank. Enter the loadname or the bank number (1 through 64) that contains the load after the **erase** command. To verify the erase, use the **list ibd** command. If an erase does not succeed after 10 tries, the device marks the bank as faulty.

IBD parameters:

IBD size: 4 MB
Bank size: 64 KB
Starting bank number: 1
Ending bank number: 64

Specifying a bank number may result in a partial erase of the load image file if it is large enough to traverse more than one bank.

Example 1:

```
erase test
Erasing bank 5 ...
Banks 1-4 contain ...
Banks 5-7 have been erased
```

Example 2:

```
erase 2
Are you sure you want to erase bank 2? (Yes or [No]): yes
Erasing bank 2 ...
Banks 5-7 has been erased
```

Example 3:

```
erase
Loadname or Bank Number: 4
Are you sure you want to erase bank 4? (Yes or [No]): yes
Erasing bank 4...
Bank 1 contains load "vl-29.cfg" which use 131094 bytes
   Loaded using TFTP over IP
   Filename config
   Host 0.0.0.0
Banks 2-3 contain load "vl-22.cfg" which uses 1832848 bytes
   Manual Booted using TKR-4/16 at (80001000, 72) as  10.1.155.29
   Filename loads/latest-gen.c5-multisna.ldc
   Host 128.185.210.125, Gateway 10.1.155.43
Bank 4 has been erased
```

If the erase fails, a message indicating the failure appears on the console along with the banks that failed. Failure information will appear in the **list** command until the router has been restarted. The router will ***not*** automatically delete any boot records referencing the image in the failed banks.

At boot time, if the boot PROM cannot find an image, it will display a message and try the next boot record.

List

Use the **list** command to display the current boot and dump configuration database, the contents of the IBD, and scheduled image load information .

Syntax: list

- addresses
- all
- boot-entries
- bp-device
- dump-entries
- ibd
- view

addresses

Displays the IP addresses and their subnet masks of all the network boot interfaces entered using the **add address** command.

Example:

```
list addresses
Interface addresses:
1: 192.9.1.1 on interface 0, mask 255.255.255.252
2: 192.9.223.39 on interface 2, mask 255.255.255.0
```

all

Displays all boot and dump configuration data and the current settings for the dump, unique-naming capabilities, and scheduled image load information .

Example:

```
Interface Addresses:

 Boot files:
  1: "/u/steve/vl/load/vl060694/vl.X11.ldc" on 216.1.2.100 via 0.0.0.

 BOOTP over interface(s): 0
 Dumping disabled
 Unique-naming disabled
 Dump to:

 Banks 1-19 contain load "vl.X11.ldc" which uses 1199272 bytes
      Loaded using TFTP over IP
      Filename /u/steve/vl/load/vl060694/vl.X11.ldc
   Host 216.1.2.100
Banks 20-48 have been erased
Bank 49 in unknown(AA) state
Banks 50-57 contain load "vl051894.ldc" which uses 508492 bytes
Loaded using TFTP over IP
Filename /u/steve/vl/load/vl051894/vl051894.ldc
   Host 216.1.2.100
Banks 58-64 have been erased

Time Activated Load Schedule Information...

 The router is scheduled to reload as follows.

 Date: April 1, 1997
 Time: 13:00
 Remote host IP address: 1.1.1.2
 Via gateway: 0.0.0.0
 Timeout in seconds: 10
 Filename: /tftpboot/vl3.img
 Interface adress: 0
 New address: 1.1.1.1
 New mask: 255.255.255.0
```

boot-entries

Displays the boot file configuration.

Example:

```
list boot-entries
1: /usr/lib/gw/this-dn.ldb on 192.9.1.2 via 0.0.0.0 for 3 secs
2: /usr/lib/gw/this.ldb on 192.9.2.2 via 192.9.1.4 for 3 secs
3: IBD load "test"
```

bp-device

Lists the interfaces that were previously added using the **add bp-device** command.

Example:

```
list bp-device
BOOTP over interface(s): 0 1
```

dump-entries

Displays the dump file configuration.

Example:

```
list dump-entries
```

ibd

Displays the contents of the IBD. It provides information similar to the GWCON **boot information** command and displays the loadname of the file and the host server from which the file was loaded. In addition, the erased and faulty banks of the IBD appear along with the faulty chips, if necessary.

Example:

```
list ibd
Bank 1 contains load "2210-29.cfg" which uses 131094 bytes
  Loaded using TFTP over IP
  Filename config
  Host 0.0.0.0
Banks 2-3 contain load "vl/load-ver2.ldc" which uses
    1652961 bytes
  Loaded using TFTP over IP
  Filename loads/vl/load-ver2.ldc
  Host 128.185.210.125
Bank 4 contains load "vl/load-ver4.cfg" which uses 131084 bytes
 Loaded using TFTP over IP
 Filename CONFIG
 Host 0.0.0.0
```

"Loaded using TFTP over IP" implies that you used the **copy** command to IBD from this local router.

view

Displays the time, date, and other information about a scheduled image load.

Example:

```
list view
 Time Activated Load Schedule Information...

  The router is scheduled to reload as follows.

  Date: April 1, 1997
  Time: 13:00
  Remote host IP address: 1.1.1.2
  Via gateway: 0.0.0.0
  Timeout in seconds: 10
  Filename: /tftpboot/vl3.img
  Interface address: 0
  New address: 1.1.1.1
  Network mask for this interface: 255.255.255.0
```

Load

Use the **load** command to copy the boot file into the device's main memory from either a local or remote source. The result of the **load** command is the same as performing the **reload** command from the * prompt.

Syntax: load local
remote

local

Retrieves a previously stored load image file from the device's IBD into the router's memory. The loadname must match one of the loadnames stored in the IBD. The loadname is case-sensitive.

To set up the IBD, use the **add boot-entry** command. This could take up to five minutes.

You must have a load file in the IBD before you can use the **load local** command successfully.

Note: If the software does not find the load file, then it will go into the boot monitor and do an auto-boot or manual boot, depending on the setting of your boot switch.

Example:

```
load local
Loadname:  ibd/softrel.ldc
```

remote

Loads the boot file from a remote host into RAM. To perform a remote load:

1. Enter the **load remote** command after the `Boot config>` prompt and enter the remote host address, remote path name, first hop address, and TFTP timeout value after the prompts.
2. A prompt then asks you to confirm the load. Enter **no** to cancel the command. Enter **yes** to load the boot file from the remote host into RAM.

Example:

```
load remote
Remote Host Address [0.0.0.0]? 128.185.210.125
Remote Pathname[]? /loads/v1.ldc
First Hop Address[0.0.0.0]? 128.185.208.38
TFTP Timeout Value [3]?
Are you sure you want to reload the gateway(Yes or No): yes
```

Remote Host Address	IP address of the host containing the boot file.
Remote Pathname	Pathname and filename of the boot file you want to load.
First Hop Address	The address of the first-hop router that routes to other networks. This is needed if the remote host address is not on a directly connected network; otherwise, use the 0.0.0.0 default.
TFTP Timeout Value	The time interval between the TFTP packet retransmissions. Longer values (longer than the default value of 3) may be needed when booting over or across slow networks or serial lines.

Store

Use the **store local** command to store a compressed image in erased banks of the IBD. The console displays the number of bytes that were stored. To verify that an image was stored, use the **list ibd** command.

Note: The router stores images sequentially from bank 1 to bank 4. When all 4 banks are full, you receive an error message. To create space in a bank, use the **erase loadname** or **erase bank-number** command.

As the device's load image file is stored into the IBD, it is compressed. The load image file will not overwrite a non-erased IBD and will not try to write beyond the end of the IBD. If the compression fails, the operator will be notified and the affected IBD will be erased.

The loadname can be any name up to 80 characters in length, can start with an alphabetic character, and is case-sensitive.

Syntax: store local . . .

loadname
Stores the specified image in an erased bank of the IBD.

Example:

```
store local
Loadname: test
Will start storing at bank #2
.
.
.
Number (dec) bytes used
Boot config>
```

Timedload

Use the **timedload** command to schedule an image load on a device, cancel a scheduled load, or to view scheduled load information.

This command allows you to load a software image into the device outside of peak network traffic periods when support personnel may not be present.

Syntax: t**imedload** activate

deactivate

view

activate Schedules an image load on the device. You will be prompted for information describing the source of the image similar to the **add boot-entry** and **add address** commands. See “Add” on page 4-12 for information about the parameters.

Time of day to load image
Specifies the date and time at which the device will load the new image. Specify the value as *YYYYMMDDHHMM*, where:

YYYY is the four-digit year.

Note: If the current month on the device is December, the year data must be the current year or the following year. Otherwise, if the current month on the device is January through November, the year data must be the current year.

MM is the two digit month.

MM Valid Values: 01 to 12 with 01 representing January.

DD is the two-digit day of the month.

DD Valid Values: 01 to 31, depending on the value of MM.

HH is the two-digit hour in 24–hour time.

HH Valid Values: 00 to 23

MM is the two-digit minute of the hour.

MM Valid Values: 00 to 59

The following are examples of scheduling a load from different sources.

Example 1. Load from a remote host:

```
Boot config> timedload activate
Time Activated Load Processing...

  Remote host IP address or IBD load name [0.0.0.0] 1.1.1.2
  Via gateway (0.0.0.0 if none) [0.0.0.0]? 0.0.0.0
  Timeout in seconds [10]? 10
  File name []? /tftpboot/vl3.cce
  Do you want to configure an interface address? (Yes, No, Quit): [No] yes
  Which interface do you want to configure an address to boot over [0]? 0
  New address [0.0.0.0]? 1.1.1.1
  Network mask for this interface [255.255.255.0]? 255.255.255.0
  Time of day to load image (YYYYMMDDHMM) []? 199703191630
  The load timer has been activated.
```

Example 2. Load from the IBD:

```
Boot config> timedload activate
Time Activated Load Processing...

  Remote host IP address or IBD load name [0.0.0.0] ibd:vl3.cce
  Time of day to load image (YYYYMMDDHMM) []? 199703191630
  The load timer has been activated.
```

deactivate

Cancels a scheduled load.

Example 1. Deactivate time activated load:

```
Boot Config> timedload deactivate
Deactivate Load Timer Processing...

  Do you want to deactivate the load timer? (Yes, No, Quit) [No]? yes
  The load timer has been deactivated
```

view Displays scheduled load information.

Example 1. Load image source is a remote host:

```
Boot Config> timedload view
Time Activated Load Schedule Information...

  The router is scheduled to reload as follows.

  Date: March 19, 1997
  Time: 16:30
  Remote host IP address: 1.1.1.2
  Via gateway: 0.0.0.0
  Timeout in seconds: 10
  Filename: /tftpboot/vl3.cce
  Interface address: 0
  New address: 1.1.1.1
  Network mask for this interface: 255.255.255.0
```

Example 2. Load image source is the IBD:

```
Boot Config> timedload view
Time Activated Load Schedule Information...

  The router is scheduled to reload as follows.

  Date: March 19, 1997
  Time: 16:30
  Filename: vl3.cce
```

TFTP

Use the **TFTP** command to initiate TFTP file transfers between a remote host and the device's non-volatile configuration memory or IBD It provides the ability to store/retrieve a load image file into/from a TFTP server or a router with an IBD.

The router acts as a TFTP client. The remote host is any device (for example, router, workstation, PC) that is running IP that acts as a TFTP server node. The router cannot be in Config-only mode.

Entering the **TFTP get** and **put** commands locks the CONFIG process for the duration of the operation. The following two keyboard character combinations are recognized during the TFTP operation:

Ctrl **P** Displays the OPCON prompt (*).
Ctrl **C** Cancels the TFTP operation.

Note: Do not press the reset button or power off the router while it is performing a **TFTP get** operation. This will leave the destination configuration memory in an inconsistent (and invalid) state. That is, you will have a partial configuration or load and it will appear to be valid.

Syntax: tftp get
put

get *CONFIG address-remote-server path/filename*

Initiates a request to a TFTP server to transfer a file *from* the server *to* the device. The server sends a data packet and the client node acknowledges receipt of the data. This cycle continues until the transfer is complete and the following message appears on the console: `TFTP transfer complete, Status: OK`

If the TFTP transfer is unsuccessful, a detailed error message appear on the screen. While transferring a file to CONFIG, the following message appears on the console: `Updating Config: Do Not Interrupt`

If you are attempting to transfer a file to IBD, and there is not enough memory in the IBD, the following message appears on the console:

```
No Free IBD Bank
```

Attention: Do not reset or power off the router while updating of the configuration memory is in progress. This may corrupt the data in configuration memory, forcing you to reconfigure the router.

Example:

```
tftp get
local filename [CONFIG]?
remote host [0.0.0.0]? 128.185.163.1
host filename [0A019947.cfg]? configs/vl-28.cfg
TFTP transfer complete, status:  OK
```

Local filename — Specifies the name that you want the file to appear under after it has been transferred to the local device. When entering the filename, make sure that you specify the **complete** pathname if you are transferring the file to the IBD. The default is CONFIG.

Remote Host — Specifies the address of the host containing the file you want to transfer. The Magic Number stored in the file is compared to the number in static RAM. This prevents cross loading non-volatile memories between types of devices.

Host filename — Specifies the name of the file on the host that you want to transfer. Make sure that you specify the **complete** pathname. The default is the ASCII representation of one of the host's IP addresses in hexadecimal. This ensures that the file has a unique name.

The hostname must match the hostname in the archive file. The hostname is case-sensitive.

put *CONFIG address-remote-server path/filename*

Initiates a request to a TFTP server to transfer a file *to* the server *from* the router. The server acknowledges the request and the client transfers the file. This cycle continues until the transfer is complete and the console displays the following message:

```
TFTP transfer complete, Status: OK
```

Note: The **TFTP put** command does not allow you to place a file in another device's configuration memory or IBD. You must be logged into that device and use the **TFTP get** command.

The console display is the same as the **TFTP get** command.

Example:

```
tftp put
Local filename [CONFIG]?
Remote host [0.0.0.0]? 128.185.163.1
Host filename [0A019947.cfg]?
TFTP transfer complete, status:  Timeout
```

local filename?	CONFIG is a filename that refers to the device's non-volatile memory.
remote Host?	You must specify the IP address of the remote host and filename to be used to store the CONFIG on the remote host.
host filename?	Specifies the name of the file on the host to which you want to transfer. Make sure that you specify the complete pathname. The default is the ASCII representation of one of the host's IP addresses in hexadecimal. This ensures that the file has a unique name. The hostname must match the hostname in the archive file. The hostname is case-sensitive.

Example:

```
tftp put IBD/r151.ldc
Remote host [0.0.0.0]? 140.187.2.100
Host filename [80B9D626.cfg]? vl605.ldc
TFTP transfer complete, status:  OK
```

To abort a TFTP transaction, press **Ctrl** **C**. Answer **yes** to `Are you sure (yes or no):`

The TFTP command generates the following error messages:

Error Message	Meaning
`Unknown Error`	Protocol failure.
`File Not Found`	Specified host file does not exist.
`Access Violation`	File protection error.
`Disk Full`	File system full during write.
`Illegal Operation`	Undefined TFTP operation requested.
`Unknown TID`	Unexpected TFTP packet received.
`File Already Exists`	File already exists.
`No Such User`	TFTP not supported on host.

Exit

Use the **exit** command to leave the current process command level and return to the previous process level.

Syntax: exit

Example: **`exit`**

Chapter 5. Boot Options

This chapter covers the boot options available. Normally, the device boots from the Integrated Boot Device (IBD). You need to use this chapter only for maintenance or diagnostic operations or for software upgrades.

The boot options allow you to boot the 2210 using the following methods:

Table 5-1. Description of Boot Methods

Boot Method	Description
IBD	Boot from the IBD using queries. Use this method when the 2210 is configured for a different boot method and you want to boot the 2210 from the IBD instead.
TFTP Host Server	Boot from a load image file on a TFTP host server. Another router can act as a TFTP host server.
BOOTP	Boot over the LAN port using the Bootstrap Protocol.

Additional options available at the boot monitor prompt let you run diagnostics, display configuration information, load configuration memory from a host on the network or through the Service port, clear configuration in SRAM, and download and upload router code through the Service port.

Included in this chapter are the following sections:

Before you Begin

Before booting the 2210, note the following:

- In order to use the procedures in this chapter, you must have a terminal connected directly to the 2210 (Refer to the *IBM 2210 Nways Multiprotocol Router Installation and Initial Configuration Guide* for an explanation of how to connect a terminal.)
- The 2210 is shipped with the boot file that is stored in the IBD.
- You cannot boot the 2210 over the ISDN interface.
- If you are booting over the Token-Ring interface and there is no Token Ring link active, you receive the following message: `lobe media test failed: function failure.`

Note: To stop a 2210 boot, press **Ctrl** **C** simultaneously on the terminal keyboard.

Booting From the Integrated Boot Device Using a Console Terminal

An example of an IBD boot using a console terminal appears at the end of this procedure. Use this boot method when you have a load image stored in the IBD.

1. The following copyright information should be on the console screen. If necessary, press the **Reset** button, then **Ctrl** **C** simultaneously to display this information.

```
 Bootstrap Monitor V1.0
(c) Copyright IBM Corp. 1994, 1997
```

2. Enter **bm** and the console displays the following information and the first boot prompt:

```
PROM Load/Dump Program * Revision: 1.0 *
Copyright IBM Corp. 1994, 1997

IBD has load(s) load image names

Device Slot Number or IBD Load Name:
```

3. Enter the load image name. The IBD load name is case-sensitive. Press **Return**. The software is loading when you see this message:

```
Loading using IBD Load Image "ibmMRNS.ldc"
```

BOOTP Using a Console Terminal

BOOTP tries to boot over all of the installed interfaces using all possible hardware configurations starting with the card that passes its self-test first. This generally occurs in the order Ethernet, and then token ring. For additional information about BOOTP, refer to Chapter 4, The Boot CONFIG Process and Commands.

A BOOTP boot is successful when the console displays the following information:

```
PROM Load/Dump Program * Revision: 1.0 *
Copyright IBM Corp. 1994, 1997

BOOTP Using interface name at (CSR address, vector address)

Trying connector
Doing BOOTP
Trying host IP address
    file name
loading
Copyright IBM Corp. 1994, 1997
Config Only Mode - Switch Selected
*
```

The * indicates that the load image has finished loading.

Unsuccessful BOOTP

A BOOTP boot fails under the following conditions:

- When the server does not know about the 2210. The console displays the following information:

```
PROM Load/Dump Program * Revision: 1.0 *
Copyright IBM Corp. 1994, 1997

BOOTP Using interface at (CSR address, vector address)

Trying connector
Doing BOOTP         BOOTP timeout
Auto BOOTP failed
```

The console then displays the prompts to perform a manual boot. Table 5-3 on page 5-5 describes these prompts.

- When the server knows about the 2210, but the load file is not present, the console displays the following information:

```
PROM Load/Dump Program * Revision: 1.0 *
Copyright IBM Corp. 1994, 1997

BOOTP Using interface at (CSR address, vector address)

Trying connector
Doing BOOTP
BOOTP got reply but server sent no filename
Manual BOOTP failed - Enter @ at prompt BOOTP again
```

Enter @ to retry BOOTP. If the retry fails, use another method to boot the 2210.

Booting from a TFTP host server using a console terminal

You can use a load image file on a TFTP host server to boot the 2210. Another router can act as a TFTP host server. An example of a TFTP boot is shown below.

1. At the boot monitor prompt, (>), enter **bm** to display the following information and the first boot prompt.

```
PROM Load/Dump Program * Revision: 1.0 *
Copyright IBM Corp. 1994, 1997

Device Types available:

               IBD
               Token Ring
               WAN
```

2. The prompts that appear depend on the type of interface you are booting over. See “BM (Boot using console queries)” on page 5-7 for details on booting an Ethernet, Token Ring, or WAN port. Table 5-3 on page 5-5 describes these prompts.

Boot Options Available

Table 5-2 on page 5-4 lists the boot options available. Detailed descriptions of the boot process and system prompts follow the table.

Accessing the Boot Options

1. Begin a load procedure by powering on the device or by typing **reload** at the OPCON (*) prompt and pressing the **Enter** key.
2. To display the Boot monitor prompt (>), press **Ctrl** **C** simultaneously during a load procedure.
3. At the boot prompt (>), enter **?** to display the boot options. Table 5-2 on page 5-4 describes these options.

Table 5-2. Boot Options

Option	Name	Description
B	Boot using stored Configuration	Boots automatically using the configuration stored in TFTP or in the IBD.
BC	Boot to Config-only Mode using console queries	Displays prompts to manually boot the 2210 and then enters Config-only mode, allowing you to begin configuring the 2210.
BM	Boot using Console Queries	Displays prompts to manually boot the 2210. Table 5-3 on page 5-5 describes these prompts.
BN	Boot, but do not run, using console queries	Used by field personnel for debugging. Boots and returns to the Bootstrap Monitor, but does not start the load.
BP	Boot using BOOTP	Displays the prompts to boot using the Bootstrap Protocol.
D	Dump using stored Configuration	This feature is not currently available on the 2210
DIAG	Initiate IBM extended diagnostics	Starts the internal tests. When internal tests are complete, you have the option of continuing with the System Extended Checkout (Internal and External Tests), the WAN/LAN Wrap Menu, or Diagnostic Utilities. You can exit and reboot at any time.
DM	Dump using Console Queries	This feature is not currently available on the 2210.
UB	Display boot Configuration	Displays the static RAM TFTP bootstrap configuration.
UC	Display Hardware Configuration	Displays the information on the hardware configuration including device types, baud rate, memory sizes, base MAC address, part numbers, serial numbers, and revision levels.
UG	Go and Execute at Address in RAM	This option is used by field service personnel.
LC	Load Configuration Memory	Loads configuration memory from a host on the network.
CC	Clear Configuration Memory	Clears the configuration in SRAM.
ZB	ZModem Boot	Downloads and uploads router code through the service port.
ZC	ZModem Configuration Memory Load	Loads configuration memory through the service port.

Boot Option Prompts

The following section explains each of the boot options in detail.

Table 5-3 on page 5-5 describes the prompts that appear when you boot the 2210. These prompts vary depending on your hardware configuration and the software loaded on the 2210.

Table 5-3 (Page 1 of 2). Boot Option Prompts

Prompt	Description
Device Type	The device type over which to boot the 2210; either the IBD, the Token-ring, or Ethernet interface.
IBD Loadname	The IBD loadname, which can include up to 79 characters, digits, and symbols and is case-sensitive. For initial installations, enter the filename in the Release Notes (file README.NTS that is on the backup software diskettes.)
Interface IP Address	The IP address of the 2210 interface over which you are booting.
IP Mask	A hexadecimal value that separates the IP network addresses from the other IP address fields. All bits that are part of the network and subnet should be 1.
Boot From Host	IP address of the host from which you are booting.
Via gateway	If the host from which you are booting is on another (sub)network, there is an intermediate router. Enter the IP address of the intermediate router.
Load Image Name	For initial installations, enter the load image name noted in the in the Release Notes (file README.NTS that is on the backup software diskettes.)
Boot File Name	Full pathname of where the load image file resides on the host server. For example, /usr/local/ibm2210.ldc (UNIX example).
Ethernet Prompts	
Connector Type (AUI/RJ45)	Enter one of the following to specify the cable type connected to this port: **AUI** Thick/AUI (10BASE5) **RJ45** Unshielded Twisted Pair (10BASE-T) **AUTOCONFIG** Automatically senses the cable type
Token Ring Prompts	
Speed (4/16)Mb	Enter **4** or **16** to represent the token ring media transfer rate in Mbps (megabits per second). **Note:** The value you enter must match the speed of the ring that you are using.
Media (UTP/STP)	Enter one of the following to specify the cable type connected to this interface: **UTP** Unshielded Twisted Pair **STP** Shielded Twisted Pair
WAN Prompts	
WAN port	WAN port over which you are booting the 2210, either **1** or **2**.
Timeout (secs)	How long, in seconds, the interface tries to boot over the network. The timeout must be greater than 5.
Clock Source (INT/EXT)	To connect to a: • Modem or DSU, enter **EXT** for external clocking. • DTE device, use a DCE cable and enter **INT** for internal clocking.
Internal Clock Speed	This prompt appears only if you enter **INT** as the Clock Source. The range is 1 to 10 000 000.

Table 5-3 (Page 2 of 2). Boot Option Prompts	
Prompt	**Description**
Cable Type (X21/Other)	Enter **X21** to connect an X.21 cable to this port. Enter **other** to connect any other cable type to this port.

B (Boot)

Boots the router automatically using the configuration stored in configuration memory. This option causes the router to boot from the IBD unless the configuration is stored on a TFTP host.

BC (Boot in Config-only Mode)

Boots the 2210 and immediately enters Config-only mode. The following examples show how to boot the 2210 over the IBD and over the Token-Ring, Ethernet, and WAN interfaces. User entries are shown in bold. To accept the defaults shown in brackets, press **Enter**.

Note: In the sample interface dialog shown below, the device's interface type appears as either Token Ring or Ethernet in the `Device Types` listing and at the `Device Type` prompt.

Enter **bc** at the boot prompt (>). The software prompts you for the following router information:

```
Device Types available:

          IBD
          Token Ring/Ethernet
          WAN
          Device Type [WAN]: IBD
```

- If you enter **IBD**, you see the following:

```
IBD has load(s) loadname
IBD Load Name:  loadname
```

  To reload the current configuration, press **Enter**.

```
Loading using IBD Load Image "load name"
```

  If you specify an incorrect or non-existent load name, the system issues the message: `No such load` and returns you to the IBD Load Name prompt.

- If you enter **Token Ring**, you see the following:

```
Media (UTP/STP) [UTP]:
Speed (4/16)Mb [16Mb]:
Interface IP address: 123.175.23.119
IP Mask (FFFFFF00):
Boot from host: 123.175.68.190
Via gateway: 123.175.23.213
Boot file name: ibmMRNS.ldc

Using Token Ring at (6000000, 0).
Trying host 123.175.68.190 via 123.175.23.213
file ibmMRNS.ldc

.loading
................................

Starting at 1040010
```

```
  The Standalone Configuration Process.  You are here because
  the watchdog timer timed out and/or Autoboot not selected.

Config (only)>
```

If there is no Token-Ring link active, you receive the following message:

```
lobe media test failed: function failure
```

- If you enter **Ethernet**, you see the following:

```
      Connector Type (AUI/RJ45)[AUTO_CONFIG]:
      Interface IP Address: 123.175.56.119
      IP Mask (FFFFFF00):
      Boot from host: 123.175.68.213
      Via Gateway: 123.175.56.190
      Boot File Name: ibmMRNS.ldc

  Using Ethernet at (6000000, 0)
  Trying host 123.175.68.213 via 123.175.56.190
  file ibmMRNS.ldc

  .loading
  ................................

  Starting at 1040010

  The Standalone Configuration Process.  You are here because
  the watchdog timer timed out and/or Autoboot not selected.

  Config (only)>
```

- Booting over a WAN

  If there is no CTS signal active on the WAN port that you specify, you will receive the following message: `CTS not active on WAN port #`

  Note: The PPP protocol is currently the only data link layer protocol that can be used when booting over a WAN interface.

BM (Boot using console queries)

Boots using console queries. The following examples show how to boot the 2210 over the IBD and over the Token Ring, Ethernet, and WAN interfaces. User entries are shown in bold. To accept the defaults shown in brackets, press **Enter**.

You can also use this option to boot from a load image file on a TFTP host server.

Note: In the sample interface dialog that follows, the interface type specific to the 2210 appears as either Token Ring or Ethernet in the `Devices Types` listing and at the `Device Type` prompt.

Enter **bm** at the boot prompt (>). The software prompts you for the following router information:

```
Device Types available:

     IBD
     Token Ring/Ethernet
     WAN

     Device Type [Token Ring/Ethernet]: IBD
```

- If you enter **IBD**, you see the following:

  ```
  IBD has load(s) load image name
  IBD Load Name: load image name
  ```

  To reload the current configuration, press **Enter**. To load another configuration, enter the load name at the prompt.

  ```
  Loading using IBD Load Image "load name"
  ```

  If you specify an incorrect or nonexistent load name, the system issues the following message: `No such load` and returns you to the IBD Load Name prompt.

- If you enter **Token Ring**, a configuration dialog similar to the following appears on your console.

  Note: If the host you specify is not directly accessible by the router, the software will prompt you to enter the IP address of the gateway. This prompt is shown below in parentheses.

  ```
  Media (UTP/STP) [UTP]:
  Speed (4/16)Mb [16Mb]:
  Interface IP address: 123.175.56.119
  IP Mask (FFFFFF00):
  Boot from host: 123.175.68.213
  Via Gateway: 123.175.56.190
  Boot File Name: ibmMRNS.ldc

Using Token Ring at (6000000, 0).
Interface configured for 16Mbps & UTP
Trying host 123.175.68.213 via 123.175.56.190
file ibmMRNS.ldc
loading
................................................

Starting at 1040000


Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
*
  ```

- If you enter **Ethernet**, you see the following:

  ```
  Connector Type (AUI/RJ45)[AUTO_CONFIG]:
  Interface IP Address: 123.175.56.119
  IP Mask (FFFFFF00):
  Boot from host: 123.175.68.213
  Via Gateway: 123.175.56.190
  Boot File Name: ibmMRNS.ldc

Using Ethernet at (6000000, 0)
  ```

```
Trying host 123.175.68.213 via 123.175.56.190
file ibmMRNS.ldc

.loading
................................

Starting at 1040000

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
*
```

- Booting over a WAN

  If there is no CTS signal active on the WAN port that you specify, you will receive the following message: `CTS not active on WAN port #`

  Note: The PPP protocol is currently the only data link layer protocol that can be used when booting over a WAN interface.

BN (Boot, But Do Not Run, Using Console Queries)

Do not use this boot option. This option is used by field service personnel only.

BP (Boot using BOOTP)

Boots using the Bootstrap Protocol. The following example shows how to boot the 2210. User entries are shown in bold. To accept the defaults shown in brackets, press **Enter**.

Note: In the following sample interface dialog, the device's interface type appears as either Token-Ring or Ethernet in the `Device Types` listing and at the `Device Type` prompt.

Enter **bp** at the boot prompt (>). The software prompts you for the following router information:

```
Device Types available:

        Token Ring/Ethernet
        Device type (for BOOTP) [Token Ring]:
```

- If you enter **Token Ring**, you see the following:

```
      Media (UTP/STP) [UTP]:
      Speed (4/16)Mb [16Mb]:

BOOTP Using Token Ring at (6000000, 0).
Doing BOOTP o
Interface configured for 16Mbps & UTP
Trying host 123.175.68.213 via 123.175.56.190
file load image name
.loading
................................................

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
```

```
*
```

- If you enter **Ethernet**, you see the following:

```
     Connector Type (AUI/RJ45)[AUTO_CONFIG]:

BootP Using Ethernet at (6000000, 0)
Doing BootP o o o o
Trying host 123.175.68.213 via 123.175.56.190
file load image name
.loading
...................

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
*
```

A BOOTP boot is successful when the terminal displays the OPCON (*) prompt.

Unsuccessful BOOTP

A BOOTP boot fails if the server is down, if the server cannot find the file you specified, or if TFTP fails. If BOOTP is unsuccessful, the terminal displays the message

```
Manual BOOTP failed - enter "@" at prompt to BOOTP again.
```

Enter @ to retry BOOTP. If the retry fails, use another method to boot the 2210.

D (Dump using stored configuration)

Writes the contents of system memory to a file when a system failure occurs. If the unique naming capability is enabled, the router automatically appends a character string to the dump filename. Using this command prevents an existing dump file from being overwritten by subsequent dumps. For information about how to enable unique naming, refer to page 4-20.

Enter **d** at the boot prompt (>). The screen displays the following information:

```
PROM Load/Dump Program * Revision 1.0
Copyright IBM Corp. 1994, 1997
Host 325.321.62.763 loading


Using Token Ring/Ethernet  (00000, 0)
Trying host 235.211.62.243 via 123.192.23.243
   file load image name

loading

Starting at 1040000
```

If the dump fails, you will receive a **Dump failed** message with a brief explanation of the cause of the failure.

DIAG (Execute IBM Extended Diagnostic Program)

Initiates internal self-test. When internal self-test is complete, you can select any of the extended diagnostics utilities provided. To run any of the extended diagnostics tests, you need the extended diagnostics Service Kit, feature code 2532. The kit includes all the necessary wrap plugs for the LAN, serial, and service ports.

1. Enter **diag** at the boot prompt (>) to execute the internal self-test. The screen displays a message similar to the following:

   ```
   Starting at 1FF00

   Starting Hardware Diagnostics
           Version:  XXXXXX  XXXXXX

   Testing System Internal

   System Checkout:  All Systems Pass

   Press space to continue.....
   ```

2. Press the space bar to get to the next level of diagnostic tests. To execute these tests you must remove the cables from the network and attach the appropriate wrap plug(s). Follow the instructions included in the extended diagnostics Service Kit for installing the wrap plugs.

   If you try to execute one of these tests without the wrap plugs installed, you receive the following message:

   ```
   You have selected a test that requires external wrap
   plugs to be present.  Remove the cable(s) from the
   network, and attach the appropriate wrap plug(s).
   ```

3. Press the space bar to select one of the diagnostic options available and follow the instructions provided with the extended diagnostics Service Kit.

   ```
   Diagnostic Main Menu (c) 1994

   1) System Checkout (Internal Tests)
   2) System Extended Checkout (Internal and External Tests)
   3) WAN/LAN Wrap Menu
   4) Diagnostic Utilities

   x) Exit (and Reboot)
   ```

DM (Dump using Console Queries)

Displays prompts to manually configure the network dump information.

Enter **dm** at the boot prompt (>).

The screen displays the following information:

```
PROM Load/Dump Program * Revision 1.0
Copyright IBM Corp. 1994, 1997
Host  ??? loading


Using Token Ring/Ethernet  (00000, 0)
Trying host 0.0.0.0 via 0.0.0.0
   file load image name
```

```
loading

Starting at 1040000
```

If the dump fails, you will receive a **Dump failed** message with a brief explanation of the cause of the failure.

UB (Display TFTP Boot Configuration)

Displays the static RAM TFTP bootstrap configuration including:

- Host name
- Whether dumping is enabled or disabled
- Whether the unique naming capability is enabled or disabled
- Interface IP address, type of interface, and mask
- Boot file name
- Host IP address
- Gateway IP address

If you have created dump files, UB also displays the dump file name and IP address of the host on which the dump files reside and the IP address of the intermediate gateway, if applicable.

To display this information: Enter **ub** at the boot prompt (>). The screen displays information similar to the example shown below.

```
TFTP bootstrap configuration:
    Host ibmMRNSV1 - .191, Dumping disabled, Unique dump naming off
Interface Addresses:
    1: 128.196.145.191 on port 0 (Token Ring/Ethernet), mask FFFFF00
Boot Files
    1: ibmMRNS.ldc on 123.175.68.213 via 123.175.56.190 for 20 secs
    2: r15.1.ldc on 123.175.68.213 via 123.175.56.190 for 20 secs
    3: ibmMRNS-univ.ldc on 123.175.68.213 via 123.175.56.190 for 20 secs
Dump Files:
    1: "gw/ibmMRNS.dmp" on 123.175.68.213 via 123.175.56.190 for 20 secs
>
```

UC (Display Hardware Configuration)

Displays the following information:

- Device types available
- Console baud rate
- Size of main memory and IBD in number of Mbytes
- Base MAC address
- Router serial number
- System card serial number
- Model number
- System card part number
- System card revision (ECO) level
- Platform revision

Note: Each 2210 is programmed at the factory with a Base MAC address in Ethernet order. If you have a Token-Ring unit, the 2210 converts the

address to Token-Ring order. However, the **uc** command displays the address in Ethernet order.

Enter **uc** at the boot prompt (>). The screen displays information similar to the following:

```
Boot device types available:
          IBD
          Token Ring
          WAN

Console Baud Rate:        9600 (Autobaud)
Main Memory size:         8 MB
IBD (flash Memory) size:  4 MB
Base MAC Address:         000093808068
System Part Number        04H7063
System Serial Number      55554000008
System EC Level           D50514
System Card Part Number   13H7771
System Card Serial Number 110653
System EC Level           C99200B
```

UG (Go execute at address in RAM)

This option is used only by your service representative.

LC (Load Configuration Memory)

Loads configuration memory from a host on the network. To use this option, do the following:

Enter **lc** at the boot prompt (>). The screen displays information similar to the following:

```
Device Types available:

          IBD
          Token Ring/Ethernet
          WAN

          Device type [Token Ring]:
```

- If you enter **Token Ring**, you will see the following:

```
        Media (UTP/STP) [UTP]:
        Speed (4/16)Mb [16Mb]:
        Interface IP address: 123.175.56.119
        IP Mask (FFFFFF00):
        Load Cfg from host: 123.175.68.213
        Via gateway: 123.175.56.190
        Config File Name: ibmMRNS.cfg

    Using Token Ring at (6000000, 0).
    Trying host 123.175.68.213 via 123.175.56.190
    file ibmMRNS.cfg

    .loading
    Receiving config memory image
    ................................................
```

```
Starting at 1040000

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
*
```

- If you enter **Ethernet**, you see the following:

```
     Connector Type (AUI/RJ45)[AUTO_CONFIG]:
     Interface IP address: 123.175.56.119
     IP mask (FFFFFF00):
     Load Cfg from host: 123.175.68.219
     Via gateway: 123.175.56.190
     Config file name: ibmMRNS.cfg

Using Ethernet at (6000000, 0).
Trying host 123.175.68.219 via 123.175.56.190
file ibmMRNS.cfg

.loading
Receiving config memory image
..............................

Starting at 1040000

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
*
```

- If you enter **WAN**, you see the following:

```
     WAN port [2]:
     Timeout (secs) [20] ?
     Clock Source (INT/EXT) [INT]:
     Internal Clock Speed 1
     Interface IP address: 123.175.56.119
     IP mask [FFFFFF00]:
     Load Cfg from host: 123.175.68.219
     Via gateway: 123.175.56.190
     Config file name: ibmMRNS.cfg

Using Serial Line at ( 0, 0).
Trying host 123.175.68.219 via 123.175.56.190
file ibmMRNS.cfg

.loading
Receiving config memory image
..............................
Starting at 1040000

Copyright Notices:
Copyright IBM Corp. 1994, 1997

MOS Operator Control
*
```

CC (Clear Configuration Memory)

Attention: Issuing this command will cause all configuration information to be lost.

This command clears the configuration in memory. Enter **cc** at the boot prompt (>). The software prompts you for basic router information as follows:

```
Are you sure you want to clear config memory?
```

ZB (ZModem Boot)

Downloads and uploads router code through the console port.

1. Enter **ZB** at the boot prompt (>) and the console displays:

   ```
   Are you sure you want to load via the console?
   ```

2. Enter **y** and the console displays the message:

   ```
   Okay, GO!!
   ```

3. Press **Return** to start the operation. The operation is completed when the system prompt (>) appears on the screen.

Note: Refer to the documentation supplied with your ZModem software for the ZModem commands to use at your console terminal.

ZC (ZModem configuration memory load)

Loads configuration memory through the console port.

Note: This option requires that the remote boot server support ZModem software.

1. Enter **ZC** at the boot prompt (>). The console displays the following prompt:

   ```
   Are you sure you want to load config memory via the console?
   ```

2. Enter **y**. The console displays the message:

   ```
   Okay, GO!!
   ```

3. Press **Return** to start the operation. The operation is completed when the boot prompt appears on the screen.

4. Enter **n** to return to the OPCON prompt.

Note: Refer to the documentation supplied with your ZModem software for the ZModem commands to use at your console terminal.

Configuring the 2210

After the 2210 has booted, you can configure it. The sections that follow briefly describe the configuration processes available when using an **ASCII terminal**.

Note: You can also use the IBM Nways Multiprotocol Routing Services Configuration Program (Configuration Program), to configure the 2210. The Configuration Program is run on a **stand-alone workstation** and has a graphical user interface. Once pre-configuration or Quick Configuration has taken place, you can use the Configuration Program to configure the 2210 completely.

Begin the configuration process as follows:

1. At the * prompt, enter **status** to display the PID (process ID) of Config.

```
Pid  Name      Status TTY  Comments
1    COpCon    RDY    TTY0
2    Monitr    DET     --
3    Tasker    RDY     --
4    MOSDDT    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    ROpCon    IDL    TTY1 128.185.133.2
8    ROpCon    RDY    TTY2 128.185.134.50
```

2. Enter **talk** and the PID. From the output in 1 on page 5-15, you would enter

   ```
   * talk 6
   ```

   Press **Return**. This displays the following information:

   ```
   Gateway user configuration
   Config>
   ```

3. You can now configure the interfaces, boot records, bridging and routing protocols using one of the following processes:

   - The **Quick Configuration Process** allows you to configure selected devices, bridging protocols, and routing protocols by responding to the Quick Configuration prompts. After creating a minimal configuration, you must transfer a complete configuration to the 2210 using TFTP.

     Enter **qc** at the `Config>` prompt to begin the Quick Configuration process.

   - **CONFIG Process** allows you to configure all bridging and routing protocols, interfaces, and boot records by entering commands at the `Config>` prompt.

     To configure the protocols using the CONFIG process, refer to the specific protocol chapters in the *Protocol Configuration and Monitoring Reference*. To configure other parameters including the interfaces and boot records, refer to the appropriate configuration chapters in this book.

Chapter 6. The GWCON (Monitoring) Process and Commands

This chapter describes the GWCON process and includes the following sections:

What is GWCON?

The Gateway Console (monitoring) process, GWCON (also referred to as CGWCON), is a second-level process of the router user interface.

Using GWCON commands, you can:

- List the protocols and interfaces currently configured in the router.
- Display memory and network statistics.
- Set current Event Logging System (ELS) parameters.
- Test a specified network interface.
- Communicate with third-level processes, including protocol environments.
- Enable and disable interfaces.

GWCON fits into the router software structure as shown in Figure 6-1.

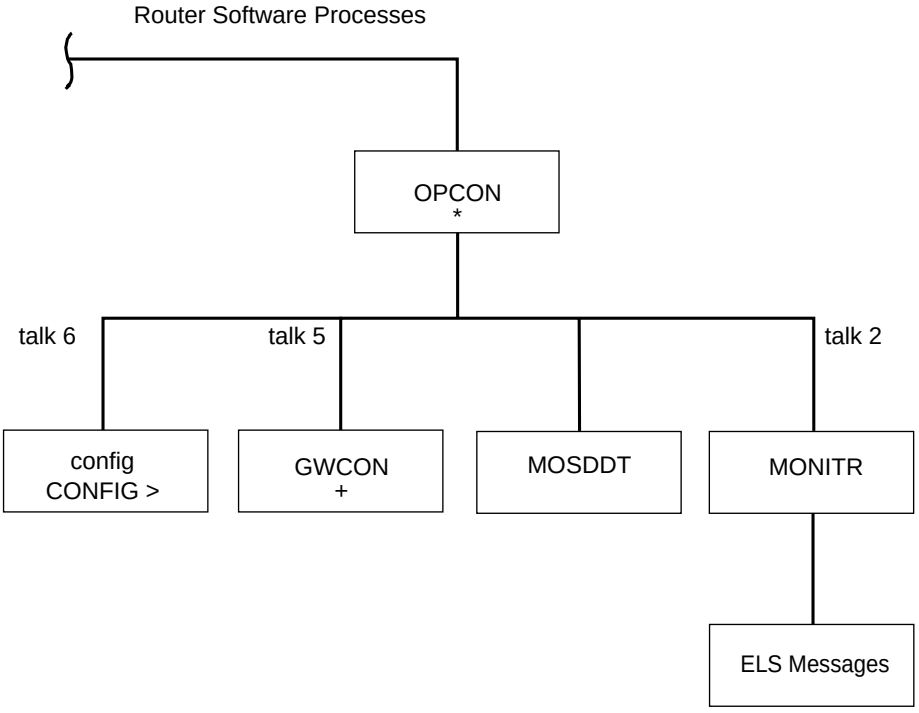


Figure 6-1. GWCON in the Router Software Structure

The GWCON command interface is made up of levels called modes. Each mode has its own prompt. For example, the prompt for the IP protocol is `IP>`.

If you want to know the process and mode you are communicating with, press **Return** to display the prompt. Some commands in this chapter, such as the

network and **protocol** commands, allow you to access the various modes in GWCON.

Entering and Exiting GWCON

To enter the GWCON command environment from OPCON and obtain the GWCON prompt:

1. At the OPCON prompt, enter the **status** command to find the process ID (PID) of GWCON.

   ```
   * status
   ```

   For sample output of the status command see “Status” on page 2-9.

2. Enter the **talk** command and the PID for GWCON to get to the GWCON prompt.

   ```
   * talk 5
   ```

The console displays the GWCON prompt (+). If the prompt does not appear, press **Return**. Now, you can enter GWCON commands.

To return to OPCON, enter the OPCON intercept character. (The default is **Ctrl** **P** .)

GWCON Commands

This section contains the GWCON commands. Each command includes a description, syntax requirements, and an example. The GWCON commands are summarized in Table 6-1.

To use the GWCON commands, access the GWCON process by entering **talk 5** and enter the GWCON commands at the (+) prompt.

Table 6-1 (Page 1 of 2). GWCON Command Summary

Command	Function
? (Help)	Lists the GWCON commands.
Activate	Enables a newly configured spare interface.
Boot	Displays information about how the device was booted last.
Buffer	Displays information about packet buffers assigned to each interface.
Clear	Clears network statistics.
Configuration	Lists status of the current protocols and interfaces.
Disable	Takes the specified interface off line.
Environment	Enters the Environment system console. Displays the current temperature and issues an alert when the temperature threshold, high or low, is passed.
Error	Displays error counts.
Event	Enters the Event Logging System environment.
Fault	Displays information about the last system fault.

Table 6-1 (Page 2 of 2). GWCON Command Summary

Command	Function
Feature	Provides access to console commands for independent router features outside the usual protocol and network interface console processes.
Interface	Displays network hardware statistics or statistics for the specified interface.
Log	Sets or views the logging level for events not included in the Event Logging System.
Memory	Displays memory, buffer, and packet data.
Network	Enters the console environment of the specified network.
Protocol	Enters the command environment of the specified protocol.
Queue	Displays buffer statistics for a specified interface.
Statistics	Displays statistics for a specified interface.
Test	Enables a disabled interface or tests the specified interface.
Uptime	Displays time statistics for the router.

? (Help)

Use the **?** (**help**) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
ACTIVATE interface
BOOT informationBUFFER statistics
CLEAR statistics
CONFIGURATION of gateway
DISABLE interface ERROR counts
ENVIRONMENT of router
EVENT logging
FAULT informationFEATURE commands
INTERFACE statistics
LOG levelMEMORY statistics
NETWORK commands
PROTOCOL commands
QUEUE lengths
STATISTICS of network
TEST network
UPTIME of gateway
```

Example:

```
protocol ?
IP
ARP
DNA
VINES
IPX
OSI
DVMRP
BGP
SNMP
OSPF
SDLC Relay
AP2
ASRT
HST
LNM
DLSW
XTP
```

Activate

Use the **activate** command to enable a spare interface on this device. See "Configuring Spare Interfaces" on page 3-7 for more information.

Syntax: activate ***interface#***

Example: **`activate 5`**

Boot

Use the **boot** command to display boot information for this device.

Syntax: boot

Example 1:

```
boot
Booted using Ethernet, line 0 at (80740000, 4) as 128.185.227.220
Filename vl.ldc
Host 128.185.122.17, Gateway 128.185.227.15
```

In the first example, the router was booted using TFTP over Ethernet. The message indicates the method of booting, the line number, the CSR (Command and Status Register) address, the IP address, the filename, the host, and the gateway. The *line number* distinguishes one port from another on a multiport board. The *CSR address* (the first of the two values in parentheses) identifies which interface board slot was used to boot the router.

The *IP address* listed after "as" (128.185.227.220 in this example) indicates which IP address the router used as its own IP address. The *Filename* is the name of the file that has the load image. The IP address listed after *Host* is the IP address of the server where the file is stored. The *Gateway*, if listed, is the router that routes the requests and responses between the server and the router that is booting.

Example 2:

```
boot
Manual Booted using Integrated Boot Device Loadname vl.ver1
```

In the second example, the router was booted manually using the Integrated Boot Device (IBD). *Manual* indicates that the boot information was entered manually at boot time.

Buffer

Use the **buffer** command to display information about packet buffers assigned to each interface.

Note: Each buffer on a device is the same size and is dynamically built. Buffers vary in size from one device to another.

To display information about one interface only, enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command.

Syntax: buffer *network#*

Example: buffer

```
              Input  Buffers:       Buffer sizes:
Nt  Interface  Req  Alloc  Low Curr  Hdr  Wrap  Data  Trail Total   Bytes Alloc
0   TKR/0       20    20     7    0  109   92    2052    7   2260   45200
1   PPP/0       20    20     7   20  109   92    2052    7   2260   45200
2   PPP/1       10    10     4    0  108   92    2048    0   2248   22480
```

Nt Network interface number associated with the software.

Interface Type of interface.

Input Buffers:

Req Number of buffers requested.

Alloc Number of buffers allocated.

Low Low water mark (flow control).

Curr Current number of buffers on this device. The value will be 0 if the device is disabled. When a packet is received, if the value of *Curr* is below *Low*, then the packet is eligible for flow control. (See the **queue** command for conditions.)

Buffer Sizes:

Hdr Sum of the maximum hardware, MAC, and data link headers.

Wrap Allowance given for MAC, LLC, or Network layer headers due to protocol wrapping.

Data Maximum data link layer packet size.

Trail Sum of the largest MAC and hardware trailers.

Total Overall size of each packet buffer.

Bytes Alloc Amount of buffer memory for this device. This value is determined by multiplying the values of *Alloc x Total*.

Clear

Use the **clear** command to delete statistical information about one or all of the router's network interfaces. This command is useful when tracking changes in large counters. Using this command does not save space or speed up the router.

Enter the interface (or net) number as part of the command. To get the interface number, use the GWCON **configuration** command.

Syntax: clear *interface#*

Example:
```
clear 1
Clear network statistics? (Yes or No):
```

Configuration

Use the **configuration** command to display information about the protocols and network interfaces. The output is displayed in three sections, the first section lists the router identification, software version, boot ROM version, and the state of the auto-boot switch. The second and third sections list the protocol and interface information.

Syntax: configuration

Example:
```
configuration


 Multiprotocol Routing Services


5765-B86 Feature 5047 V1 R1.0 PTF 0 RPQ 0
Boot ROM version  1.20     Watchdog timer enabled  Auto-boot enabled
Time: 15:46:12    Friday     September 20, 1996  Console baud rate: 9600

Num Name  Protocol
0   IP    DOD-IP
3   ARP   Address Resolution
11  SNMP  Simple Network Management Protocol
12  OSPF  Open SPF-Based Routing Protocol
23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge
26  DLS   Data Link Switching

Num Name  Feature
2   MCF   MAC Filtering

3 Networks:
Net Interface  MAC/Data-Link         Hardware          State
0   TKR/0   Token-Ring/802.5         Token-Ring         Up
1   Eth/0   Ethernet/IEEE 802.3      Ethernet/802.3     Up
2   PPP/0   Point to Point           SCC Serial Line    Up
```

- The first line gives the product name.
- The second line lists the program/product number, Feature Number, Version, Release, PTF and RPQ information.
- The third line displays the version of the Boot PROM (Programmable Read Only Memory) that is currently installed in the router, and the current settings of the Watchdog Timer and Autoboot switches.
- The fourth line displays the date and time, and the current console baud rate settings for DTE and DCE, respectively.

- The remaining lines list the configured protocols, followed by the configured features.

Protocols:

Num
Number that is associated with the protocol.

Name
Abbreviated name of the protocol.

Protocol
Full name of the protocol.

Features:

Num
Number associated with the feature.

Name
Abbreviated name of the feature.

Feature
Full name of the feature.

Networks:

Net Network number that the software assigns to the interface. Networks are numbered starting at 0. These numbers correspond to the interface numbers discussed under the CONFIG process.

Interface
Name of the interface and instance of this type of interface.

MAC/Data Link
Type of MAC/Data link configured for the interface.

Hardware
Specific kind of interface by hardware type.

State
Current state of the network interface.

Testing Indicates that the interface is undergoing a self-test. Occurs when the router is first started, when a problem is detected on the interface, or when the **test command** is used.

When an interface is operational, the interface periodically sends out maintenance packets and/or checks the physical state of the port or line to ensure that the interface is still functioning correctly. If the maintenance fails, the interface is declared down and a self-test is scheduled to run in 5 seconds. If a self-test fails, the interface transitions to the down state and the interval until the next self-test is increased up to a maximum of 2 minutes. If the self-test is successful, the network is declared up.

Up Indicates the interface is operational.

Down Indicates that the interface is not operational and has failed a self-test. The network will periodically transition to the testing state to determine if the interface can become operational again.

Disabled

Indicates that the interface is disabled. An interface can be disabled by the following methods:

- An interface can be configured as disabled using the CONFIG **disable** command. Each time the router is reinitialized, the interface's initial state will be disabled. It will remain in the disabled state until an action is taken to enable it.
- An interface can be disabled using the GWCON **disable** command. This method is temporary because the interface will revert to its configured state (enabled or disabled) when the router is reinitialized.
- The network manager can disable the interface through SNMP. This method is temporary because the interface will revert to its configured state (enabled or disabled) when the router is reinitialized.

When an interface is disabled, it remains disabled until one of the following methods is used to enable it:

- The GWCON **test** command is used to start a self-test of the interface.
- The network manager initiates a self-test of the interface through SNMP.

WAN Reroute also can change the state of a disabled interface. If an interface is configured as an alternate interface for WAN Reroute and its configured state is disabled, WAN Reroute will start a self-test of the interface when the primary interface goes down. When the primary interface is operational and stable again, WAN Reroute puts the alternate interface back to its configured state. Refer to Chapter 16, “The WAN Reroute Feature” on page 16-1 for more information.

Available

Indicates that the interface has been configured as a secondary WAN Restoral interface and it is available to back up the primary interface.

Not Present

Indicates that the interface’s adapter is not plugged in.

Not Present is also used as the state for a null device. Spare interfaces are displayed as null devices until they are activated.

HW Mismatch

Indicates that the configured adapter type does not match the adapter type that is actually present in the slot.

Disable

Use the **disable** command to take a network interface off-line, making the interface unavailable. This command immediately disables the interface. You are not prompted to confirm, and no verification message displays. If you disable an interface with this command, it remains disabled until you use the GWCON **test** command or an OPCON **restart** or **reload** command to enable it.

Enter the interface, or net number as part of the command. To obtain the interface number , use the GWCON **configuration** command.

Note: If the interface you are disabling is configured as an alternate WAN Reroute interface, you are asked if you want to disable any WAN Reroute primary/alternate pairings that include this alternate interface. If you answer *yes*, the interface is disabled and is no longer available to backup a primary interface. If you answer *no*, the alternate interface is disabled but WAN Reroute will attempt to bring it up if its corresponding primary interface goes down. See Chapter 16, “The WAN Reroute Feature” on page 16-1, Chapter 14, “Configuring WAN Restoral” on page 14-1, and Chapter 15, “Monitoring WAN Restoral” on page 15-1 for additional information.

Syntax: d**isable** *interface#*

Example: **`disable interface 1`**

Environment

Note: Invoke this command ***only*** for routers with two service ports.

Displays the `ENV>` prompt, which has three available commands: **list, reset-max-min,** and **exit**. Type **exit** to return to the + prompt.

In extreme temperature conditions, the temperature chip holds the router in a reset state, preventing it from operating. To ensure correct operation of the router due to temperature conditions, the temperature chip allows the router to operate in the range –55°C to 85°C. This is not the operational range.

The temperature chip shuts off the router at 85°C (185°F) or above and does not come back on until it is 80°C (176°F) or below. Only heat affects the chip. It does not cause the router to reset on cold conditions. Minus 55°C (–67°F) is the lowest temperature the chip registers.

Syntax: environment

Example:

```
environment
Environment System user console
ENV>
```

The **list** command displays a status screen with the current temperature, the amount of time between successive temperature readings, the noted maximum and minimum seen since the last reset/clear, and alerts when the temperature threshold, high or low, has been passed, as well as the hysteresis value.

Example: `list`

```
Time: 14:23:12    Sunday, January 09 2011

Current Ambient Temperature: 44C (111F)

Recalculate temperature approx. every 60 seconds.

Maximum:  48C (118F) at 11:47:32   Friday,     January 07 2011
Minimum:  40C (104F) at 15:24:21   Saturday,   January 08 2011
Last Max/Min Reset:     09:21:17   Thursday,   January 06 2011

High Temperature Alarm Threshold:  85C (185F)
Low Temperature Alarm Threshold:   -55C (-67F)
 (Hysteresis value: +/- 5C))
```

The **reset-max-min** command sets the value of the last recorded maximum and minimum to the current temperature. This is similar to resetting a standard high-low thermometer.

Example `reset-max-min`

```
Maximum and Minimum Temperature reset to current ambient temperature: 44C (111F)
```

Error

Use the **error** command to display error statistics for the network. This command provides a group of error counters.

Syntax: error

Example: **`error`**

```
                        Input   Input      Input      Input     Output  Output
Nt  Interface  Discards  Errors  Unk Proto  Flow Drop   Discards  Errors
0   TKR/0             0       0          0          0          0          0
1   PPP/0             0       0          0          0          0          0
2   PPP/1             0       0          0          0          0          0
```

Nt Network interface number associated with the software.

Interface Type of interface.

Input Discards Number of inbound packets which were discarded even though no errors were detected to prevent their being deliverable to a higher-layer protocol. The packets may have been discarded to free buffer space.

Input Errors Number of packets that were found to be defective at the data link.

Input Unk Proto Number of packets received for an unknown protocol.

Input Flow Drop Number of packets received that are flow controlled on output.

Output Discards Number of packets that the router chose to discard rather than transmit due to flow control.

Output Errors Number of output errors, such as attempts to send over a network that is down or over a network that went down during transmission.

Note: The sum of the discarded output packets is not the same as input flow drops over all networks. Discarded output may indicate locally originated packets.

Event

Use the **event** command to access the Event Logging System (ELS) console environment. This environment is used to set up temporary message filters for troubleshooting purposes. All changes made in the ELS console environment will take effect immediately, but will go away when the router is reinitialized. See Chapter 8, “Using and Configuring the Event Logging System (ELS)” on page 8-1 for information about the Event Logging System and its commands. Use the **exit** command to return to the GWCON process.

Syntax: event

Example: `event`

Fault

Use the **fault** command to display information about the last system fault. This diagnostic information can help your service representative trace recurring system errors. Output that is generated is for use by the service representative only.

Syntax: fault

Example: `fault`

Feature

Use the **feature** command to access console commands for specific 2210 features outside of the protocol and network interface console processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release. For example:

```
feature ?
```

To access that feature's console prompt, enter the **feature** command at the GWCON prompt followed by the feature number or short name. Table 3-4 on page 3-26 lists available feature numbers and names.

Once you access the prompt for that feature, you can begin entering specific commands to monitor that feature. To return to the GWCON prompt, enter the **exit** command at the feature's console prompt.

Syntax: feature *feature# OR feature-short-name*

Example:

Interface

Use the **interface** command to display statistical information about the network interfaces (for example, Ethernet or Token-Ring). This command can be used without a qualifier to provide a summary of all the interfaces (shown in the following output) or with a qualifier to reveal detailed information about one specific interface.

Descriptions of detailed output for each type of interface are provided in the specific interface *Monitoring* chapters found in this guide. To obtain the interface number, use the GWCON **configuration** command.

Syntax: interface *interface#*

Example: `interface`

```
                                    Self-Test  Self-Test  Maintenance
 Nt Nt' Interface      CSR  Vec      Passed     Failed       Failed
 0  0   Eth/0        81600   5E           1          0            0
 1  1   PPP/0        81620   5D           0         31            0
 2  2   PPP/1        81640   5C           0         31            0
```

Note: The display varies depending on the device.

Nt Global interface number.

Nt' Reserved for dial circuit use. Interface number of the physical network interface that the dial circuit uses.

Interface Interface name.

CSR Command and Status Register address.

Vec Interrupt vector.

Self-Test Passed
Number of times self-test succeeded (state of interface changes from down to up).

Self-Test Failed
Number of times self-test failed (state of interface changes from up to down).

Maintenance Failed
Number of maintenance failures.

Log

Use the **log** command to view or temporarily change the current logging level of messages that are not included in the Event Logging System. The command is temporary and goes away when the router is reinitialized.

To display the current logging level, do not enter an octal number as part of the command. To change the logging level, enter the octal number of the new logging level as part of the command. The default logging level is 76 (octal).

Note: To change the initial logging level (that is, the level that the router uses when it starts), use the CONFIG **set logging level** command. (Refer to Chapter 3, "The CONFIG Process and Commands" on page 3-1 for information about this command.)

Syntax: log [*octal_#*]

Example: **`log`**

```
Log lvl:  76
```

Memory

Use the **memory** command to display the current CPU memory usage in bytes, the number of buffers, and the packet sizes.

To use this command, free memory must be available. The number of free packet buffers may drop to zero, resulting in the loss of some incoming packets; however, this does not adversely affect router operations. The number of free buffers should remain constant when the router is idle. If it does not, contact your service representative.

Syntax: memory

Example: **`memory`**

```
                Total  Reserve    Never      Perm     Temp     Prev
                                  Alloc     Alloc    Alloc    Alloc

Heap memory    5463895   201824  5065383    328344   375856    22656

Number of global buffers:  Total = 294, Free = 287, Fair = 57, Low = 58
Global buff size:  Data = 4478, Header = 128, Wrap = 92, Trailer = 19
Total = 4700
```

Heap memory: Amount of memory used to dynamically allocate data structures.

Total	Total amount of space available for allocation for memory.
Reserve	Minimum amount of memory needed by the currently configured protocols and features.
Never Alloc	Memory that has never been allocated.
Perm Alloc	Memory requested permanently by router tasks.
Temp Alloc	Memory allocated temporarily to router tasks.
Prev Alloc	Memory allocated temporarily and returned.
Number of global buffers:	
Total	Total number of global buffers in the system.
Free	Number of global buffers available.
Fair	Fair number of buffers for each interface. (See “Low.”)
Low	The number of free buffers at which the allocation strategy changes to conserve buffers. If the value of *Free* is less than *Low*, then buffers will not be placed on any queue that has more than the *Fair* number of buffers in it.
Global buff size:	Global buffer size.
Data	Maximum data link packet size of any interface.
Header	Sum of the maximum hardware, MAC, and data link headers.
Wrap	Allowance given for MAC, LLC, or Network layer headers due to protocol wrapping.
Trailer	Sum of the largest MAC and hardware trailers.
Total	Overall size of each packet buffer

Network

Use the **network** command to enter the console environment for supported networks, such as X.25 networks. This command obtains the console prompt for the specified interface. From the prompt, you can display statistical information, such as the routing information fields for Token-Ring networks.

At the GWCON prompt (+), enter the **configuration** command to see the protocols and networks for which the router is configured. See “Configuration” on page 6-6 for more information on the configuration command.

Enter **interface** at the + prompt for a display of the networks for which the router is configured.

Enter the GWCON **network** command and the number of the interface you want to monitor. For example:

```
+network 3
X.25>
```

In the example, the X.25> prompt is displayed. You can then view information about the X.25 interface by entering the X.25 monitoring commands.

Exiting the Interface Console Process

To exit the interface console process and return to the OPCON process:

1. Return to the GWCON process by entering the **exit** command. For example:

   ```
   >X.25>exit
   ```

2. Return to the OPCON process by entering the OPCON intercept character. (**Ctrl** **P** is the default intercept character.)

After identifying the interface number of the interface you want to monitor, for interface-specific information, see the monitoring chapter in this manual for the specified network or link-layer interface. Console support is offered for the following network and link-layer interfaces:

- Ethernet
- Frame Relay
- PPP
- SDLC
- SDLC Relay (SRLY)
- Token Ring
- V.25bis
- X.25
- ATM
- ISDN
- V.34
- Dial-In
- Dial-Out
- Multilink PPP (MP)

Syntax: network *interface#*

Example:

```
network 2
ETH>
```

Protocol

Use the **protocol** command to communicate with the router software that implements the network protocols installed in your router. The **protocol** command accesses a protocol's command environment. After you enter this command, the prompt of the specified protocol appears. From the prompt, you can enter commands that are specific to that protocol.

Enter the protocol number or short name as part of the command. To obtain the protocol number or short name, enter the CONFIG command environment (`Config>`), and then enter the **list configuration** command. See "Entering and Exiting CONFIG" on page 3-10 for instructions on accessing `Config>`. To return to GWCON, enter **exit**.

See the corresponding monitoring chapter in this manual or in the *Protocol Configuration and Monitoring Reference* for information on a specific protocol's console commands.

Syntax: protocol *prot#*

Example:

```
protocol 7
IPX>
```

The following table lists examples of protocol numbers and names.

Protocol Number	Protocol Short Name	Accesses the following protocol process
0	IP	IP (Internet Protocol)
3	ARP	ARP (Address Resolution Protocol)
4	DN	DNA Phase IV
6	VIN	Banyan VINES
7	IPX	IPX (Novell NetWare Internetwork Packet Exchange)
8	OSI	ISO CLNP/ESIS/ISIS
9	DVMRP	Distance Vector Multicast Routing Protocol
10	BGP	Border Gateway Protocol
11	SNMP	SNMP (Simple Network Management Protocol)
12	OSPF	OSPF (Open Shortest Path First)
20	SDLC	SDLC Relay
22	AP2	AppleTalk Phase 2
23	ASRT	Adaptive Source Routing Transparent Bridge
24	HST	TCP/IP Host Services
25	LNM	LAN Network Manager
26	DLSW	Data Link Switching
27	XTP	X.25 Transport Protocol
28	APPN HPR	APPN High Performance Routing
30	APPN ISR	APPN Intermediate Session Routing

Queue

Use the **queue** command to display statistics about the length of input and output queues on the specified interfaces. Information about input and output queues provided by the queue command includes:

- The total number of buffers allocated
- The low-level buffer value
- The number of buffers currently active on the interface.

To display information about one interface only, enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command.

Syntax: queue *interface#*

Example: **queue**

```
                    Input Queue       Output Queue
Nt Interface  Alloc Low Curr    Fair Curr
0  Eth/0         30  10   30      30    1
1  PPP/0         24   4   24       4    0
2  FR/0          24   4   24       5    0
```

Nt Network interface number associated with the software.

Interface	Type of interface.
Input Queue:	
Alloc	Number of buffers allocated to this device.
Low	Low water mark for flow control on this device.
Curr	Current number of buffers on this device. The value will be 0 if the device is disabled.
Output Queue:	
Fair	Fair level for the length of the output queue on this device.
Curr	Number of packets currently waiting to be transmitted on this device. For locally originated packets, the eligibility discard depends on the global low water mark described in the **memory** command.

The router attempts to keep at least the Low value packets available for receiving over an interface. If a packet is received and the value of Curr is less than Low, then the packet will be subject to flow control. If a buffer subject to flow control is to be queued on this device and the Curr level is greater than Fair, then the buffer is dropped instead of queued. The dropped buffer is displayed in the Output Discards column of the **error** command. It will also generate ELS event GW.036 or GW.057.

Due to the scheduling algorithms of the router, the dynamic numbers of Curr (particularly the Input Queue Curr) may not be fully representative of typical values during packet forwarding. The console code runs only when the input queues have been drained. Thus, Input Queue Curr will generally be nonzero only when those packets are waiting on slow transmit queues.

Statistics

Use the **statistics** command to display statistical information about the network software, such as the configuration of the networks in the router.

To display information about one interface only, enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command.

Syntax: statistics *interface#*

Example: **`statistics`**

```
                Unicast  Multicast      Bytes    Packets      Bytes
 Nt Interface  Pkts Rcv   Pkts Rcv   Received      Trans      Trans
 0  Eth/0           137          1       8832       1068      65297
 1  PPP/0             0          0          0          0          0
 2  PPP/1             0          0          0          0          0
```

Nt Network interface number associated with the software.

Interface Type of interface.

Unicast Pkts Rcv
Number of non-multicast, non-broadcast specifically-addressed packets at the MAC layer.

Multicast Pkts Rcv
Number of multicast or broadcast packets received.

Bytes Received
: Number of bytes received at this interface at the MAC layer.

Packets Trans
: Number of packets of unicast, multicast, or broadcast type transmitted.

Bytes Trans
: Number of bytes transmitted at the MAC layer.

Test

Use the **test** command to verify the state of an interface or to enable an interface that was previously disabled with the **disable** command. If the interface is enabled and passing traffic, the **test** command will remove the interface from the network and run self-diagnostic tests on the interface.

Enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command. When testing starts, the console displays the following message:

```
Testing net 0 TKR/0...
```

When testing completes or fails, or when GWCON times out (after 30 seconds), the following possible messages are displayed:

```
Testing net 0 Eth/0 ...successful

Testing net 0 Eth/0 ...failed
Testing net 0 Eth/0 ...still testing
```

Some interfaces may take more than 30 seconds before testing is done.

Note: If the interface you are testing is configured as an alternate WAN Reroute interface, you are prompted:

- If you want to enable the interface's primary-alternate pairings if WAN Reroute is currently disabled for the alternate interface.

  If you answer *yes*, the same action occurs as when you enter the **t 5** `WAN Reroute>` **enable alternate-circuit** command described in Chapter 15, “Monitoring WAN Restoral” on page 15-1.
- If you want to test the interface.

  Normally an alternate WAN Reroute interface is disabled until it is needed to back up its corresponding primary interface. If you answer *yes*, a self-test is started for the interface. If you answer *no*, a self-test does not occur.

See Chapter 16, “The WAN Reroute Feature” on page 16-1, Chapter 14, “Configuring WAN Restoral” on page 14-1, and Chapter 15, “Monitoring WAN Restoral” on page 15-1 for additional information.

Syntax: test *interface#*

Example: **`test 0`**

Note: For this command to work, you must enter the ***complete*** name of the command followed by the interface number.

Uptime

Use the **uptime** command to display time statistics about the router, including the following:

- Number of restarts.
- Number of known crashes.
- Whether the router was last reloaded or restarted.
- Time elapsed since the last reload.
- Time elapsed since the last restart.

Syntax: uptime

Example: **uptime**

```
1 start, (0 known crashes) Last: Reloaded

Last Reload:  4 hours, 46 minutes ago
Last Restart:  4 hours, 46 minutes ago
```

Chapter 7. The MONITR Process

This chapter explains how to use the MONITR process and how to control the way MONITR collects and displays messages. (Refer to Chapter 8, “Using and Configuring the Event Logging System (ELS)” on page 8-1 for information about ELS and message formats. Refer also to the *IBM Nways Event Logging System Messages Guide* for a description of each message. This chapter includes the following sections:

- “What is MONITR?”
- “Commands Affecting MONITR”
- “Entering and Exiting MONITR” on page 7-2
- “Receiving MONITR Messages” on page 7-2

What is MONITR?

The MONITR process provides a view of activity inside the router and the networks. MONITR also displays logging messages from software that still uses the old logging system. MONITR fits into the router software structure as shown in 7-1.

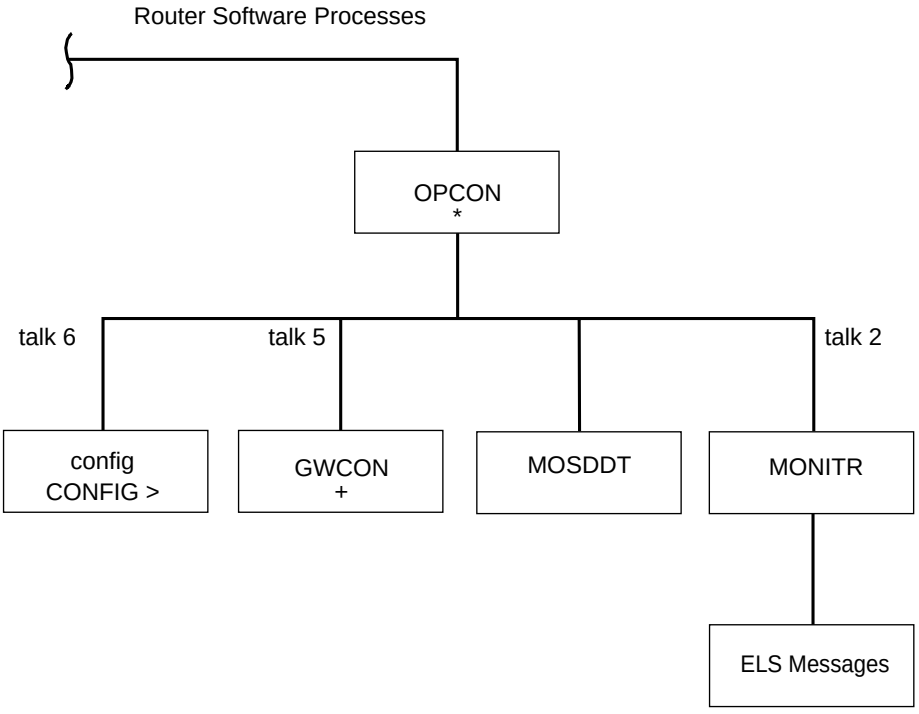


Figure 7-1. MONITR in the Router Software Structure

Commands Affecting MONITR

The following commands affect the MONITR process:

- OPCON commands:
  - **divert** temporarily diverts output to a different device.
  - **flush** causes MONITR to discard the messages it collects.

  - **halt** reverses the action of the divert command.
  - **talk** causes MONITR to display its output.
- CONFIG **set logging disposition** command sets the initial device to which MONITR sends its output.

Entering and Exiting MONITR

To enter the MONITR process from OPCON:

1. At the OPCON prompt, enter the status command to find the PID (process ID) of MONITR.

   ```
   * status
   ```

2. Enter the **talk** command and the PID number to enter the MONITR environment.

   ```
   * talk 2
   ```

MONITR does not display any prompt and you cannot enter any commands; however, the console begins to display the messages MONITR has accumulated.

To exit MONITR and return to OPCON, enter the OPCON intercept character (the default is **Ctrl** **P**).

Receiving MONITR Messages

To receive MONITR messages at your console, contact MONITR as described in the previous section. Then MONITR displays all the messages it has recorded since it was last invoked. While you are connected to MONITR, it displays all messages as they arrive.

You can use the OPCON **divert** and **halt** commands to view MONITR messages while you are doing something else with the router. Permitted devices divert output to TTY0 (the local console), TTY1, or TTY2 (the remote consoles).

To specify a default device for MONITR, define the device in Static RAM by using the CONFIG **set logging disposition** command. Specifying a default device is useful if you have a terminal set up to print.

Chapter 8. Using and Configuring the Event Logging System (ELS)

This chapter describes the Event Logging System (ELS) and its configuration. The ELS continually logs all events, filtering them according to parameters that you select. A combination of the GWCON counters and the ELS provides information for monitoring the health and activity of the system. The information is divided into the following sections:

What is ELS?

ELS is a monitoring system and an integral part of the router operating system. ELS manages the messages logged as a result of router activity. Using ELS commands, you can set up a configuration that sorts out only those messages that are important to you. You can display the messages on the console terminal screen or send the messages to a network management station using Simple Network Management Protocol (SNMP) traps.

The ELS system and the GWCON counters are the best troubleshooting tools you have to isolate problems in the router. A quick scan of the event messages will tell you whether or not the router has a problem and basically where to start looking for it.

In the ELS configuration environment, the commands are used to establish a default configuration. This default configuration does not take effect until you reinitialize the router.

Occasionally, it is necessary to temporarily view messages other than what was set up in the ELS configuration environment without having to reinitialize the router. The ELS console environment is used to:

- Temporarily change the default ELS display settings
  - Changes made in the ELS console environment take effect immediately
  - Changes made using console commands are not stored in nonvolatile configuration storage.
- View statistical information regarding ELS uses of dynamic RAM

Note: Specific ELS messages are described in the *IBM Nways Event Logging System Messages Guide*

ELS is a subprocess that you access from the

OPCON process. ELS fits into the router software structure as shown in Figure 8-1 on page 8-2.

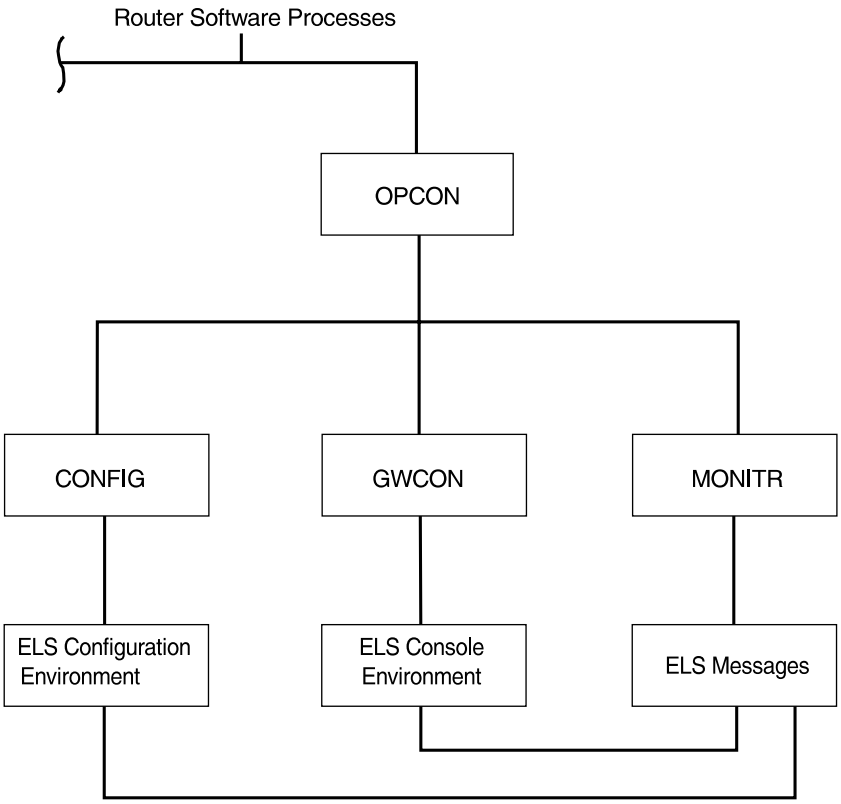


Figure 8-1. ELS in the Router Software Structure

Entering and Exiting the ELS Configuration Environment

The ELS configuration environment (available from the CONFIG process) is characterized by the `ELS Config>` prompt. Commands entered at this prompt create the ELS default state that takes effect after you restart the router. These commands are described in greater detail later in this chapter.

Configuration commands that have subsystem, group, or event as a parameter are executed in the following order:

- Subsystem
- Group
- Event

To set a basic ELS configuration, enter the **display subsystem all standard** command at the `ELS Config>` prompt. This command configures the ELS to display messages from all subsystems with the STANDARD logging level (that is, all errors and unusual informational comments).

Note: The router does not have a default ELS configuration. You must enter the ELS configuration environment and set the default state.

This section describes how to enter and exit the ELS configuration and console environments.

ELS Configuration Environment

To enter the ELS configuration environment from OPCON:

1. At the OPCON prompt, enter the **status** command to find the PID (process ID) of CONFIG.

   ```
   * status
   ```

2. Enter the **talk** command and the PID for CONFIG.

   ```
   * talk 6
   ```

   The console displays the CONFIG prompt (`Config>`). If the prompt does not appear when you first enter CONFIG, press **Return**.

3. At the CONFIG prompt, enter the following command to access ELS:

   ```
   Config> eve
   ```

   The console displays the ELS configuration prompt (`ELS config>`). Now, you can enter ELS configuration commands.

To leave the ELS configuration environment, enter the **exit** command. This command is described in this chapter.

Event Logging Concepts

This section describes how events are logged and how to interpret messages. Also described are the concepts of subsystem, event number, and logging level. A large part of ELS function is based on commands that take the subsystem, event number, and logging level as parameters.

Causes of Events

Events occur continuously while the router is operating. They can be caused by any of the following reasons:

- System activity
- Status changes
- Service requests
- Data transmission and reception
- Data and internal errors

When an event occurs, ELS receives data from the system that identifies the source and nature of the event. Then ELS generates a message that uses the data received as part of the message.

Interpreting a Message

This section describes how to interpret a message generated by ELS. Figure 8-2 shows the message contents.

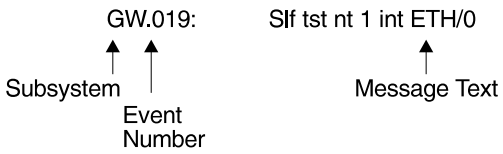


Figure 8-2. Message Generated by an Event

The information illustrated in Figure 8-2 as well as the ELS logging level information displayed with the **list subsystem** command is as follows:

Subsystem

Subsystem is a predefined short name for a router component, such as a protocol or interface. In Figure 8-2 on page 8-3, **GW** identifies the subsystem through which this event occurred.

Other examples of subsystems include IP, TKR, and X25. On a particular router, the actual subsystems present depend on the hardware and software configured for that router. You can use the **list subsystem** command described in this chapter to see a list of the subsystems on your router.

Enter the subsystem as a parameter to an ELS command when you want the command to affect the entire subsystem. For example, the ELS command **display subsystem GW** causes all events (except the events with 'debug' logging level) that occur through the GW subsystem to be displayed.

Event Number

Event Number is a predefined, unique, arbitrary number assigned to each message within a subsystem. In Figure 8-2 on page 8-3, **19** is the event number within the GW subsystem. You can see a list of all the events within a subsystem by using the **list subsystem** command, where *subsystem* is the short name for the subsystem.

The event number always appears with a subsystem, separated by a period. For example: **GW.019**. The subsystem and event number together identify an *individual* event. They are entered as a parameter to certain ELS commands. When you want a command to affect only the specified event, enter the subsystem and event number as a parameter for the ELS command.

Logging Level

Logging level is a predefined setting that classifies each message by the type of event that generated it. This setting is displayed whenever you use the **list subsystem** ELS console command. Table 8-1 on page 8-5 lists the logging levels and types.

Table 8-1. Logging Levels

Logging Level	Type
UI ERROR	Unusual internal errors
CI ERROR	Common internal errors
UE ERROR	Unusual external errors
CE ERROR	Common external errors
ERROR	Includes all error levels above
UINFO	Unusual informational comment
CINFO	Common informational comment
INFO	Includes all comment levels above
STANDARD	Includes all error levels and all informational comment levels (default)
PTRACE	Per packet trace
UTRACE	Unusual operation Trace message
CTRACE	Common operation Trace message
TRACE	Includes all trace levels above
DEBUG	Message for debugging
ALL	Includes all logging levels

In Table 8-1, ERROR, INFO, TRACE, STANDARD, and ALL are aggregates of other logging level types. STANDARD is the recommended default.

The logging level setting affects the operation of the following commands:

- **Display subsystem**
- **Nodisplay subsystem**
- **Trap subsystem**
- **Notrap subsystem**

The logging level is set for a particular command when you specify it as a parameter to one of the above commands. For example:

```
display subsystem TKR ERROR
```

Including the logging level on the command line modifies the **display** command so that whenever an event with a logging level of either UI-ERROR or CI-ERROR occurs through subsystem TKR, the console displays the resulting message.

You cannot specify the logging level for operations affecting groups or events.

Message Text

Message Text appears in short form. In Figure 8-2 on page 8-3, `Slf tst nt 1 int ETH/0` is the message generated by this event. Variables, such as *source_address* or *network*, are replaced with actual data when the message displays on the console.

The variable *error_code* is referred to by some of the Event Logging System message descriptions (usually preceded by rsn or reason). They indicate the type of packet error detected. Table 8-2 on page 8-6 describes the error or packet

completion codes. Packet completion codes indicate the disposition of the packets that arrive at the router.

Table 8-2. Packet Completion Codes (Error Codes)

Code	Meaning
0	Packet successfully queued for output
1	Random, unidentified error
2	Packet not queued for output due to flow control reasons
3	Packet not queued because network is down
4	Packet not queued to avoid looping or bad broadcast
5	Packet not queued because destination host is down (only on networks where this can be detected)

ELS displays network information as follows:

`nt 1 int Eth/0` (or) `network 1, interface Eth/0,`

where:

- 1 is the network number (each network on the router is numbered sequentially from zero).
- 0 is the unit number (the interfaces of each hardware type are numbered sequentially from zero).

Ethernet and 802.5 hardware addresses appear as a long hexadecimal number.

IP (Internet Protocol) addresses are printed as 4 decimal bytes separated by periods, such as 18.123.0.16.

Groups

Groups are user-defined collections of events that are given a name, the group name. Like the subsystem, subsystem and event number, and logging level, you can use the group name as a parameter to ELS commands. However, there are no predefined group names. You must create a group before you can specify its name on the command line.

To create a group, use the **add** configuration command described in this chapter, specify the name you want to call the group, and then specify the events you want to be part of the group. The events you add to the group can be from different subsystems and have different logging levels.

After creating a group, you can use the group name to manipulate the events in the group as a whole. For example, to turn off display of all messages from events that have been added to a group named grouptwo, include the group name on the command line, as follows:

```
nodisplay group grouptwo
```

To delete a group, use the **delete** command.

ELS Configuration Commands

Table 8-3 summarizes the ELS configuration commands. The remainder of this section describes each one in detail. After accessing the ELS configuration environment, you can enter ELS Configuration commands at the `ELS Config>` prompt.

Table 8-3. ELS Configuration Command Summary

Command	Function
? (Help)	Lists the ELS configuration commands or lists the options associated with specific commands.
Add	Adds an event to an existing group or creates a new group.
Clear	Clears all ELS configuration information.
Default	Resets the display or trap setting of an event, group, or subsystem.
Delete	Deletes an event number from an existing group or deletes an entire group.
Display	Enables message display on the console monitor.
Exit	Exits the ELS configuration process and returns you to the CONFIG process.
List	Lists information on ELS settings and messages.
Nodisplay	Disables message display on the console.
Notrace	Controls disablement of packet trace events.
Notrap	Keeps messages from being sent out in SNMP traps.
Set	Sets the pin parameter, the timestamp feature, and ATM packet tracing options.
Trace	Controls enablement of packet trace events.
Trap	Allows messages to be sent to a network management workstation in SNMP traps.
View	Allows viewing of traced packets.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: *?*

Example: **?**

```
ADD
CLEAR
DEFAULT
DELETE
DISPLAY
LIST
NODISPLAY
NOTRACE
NOTRAP
SET
TRACE
TRAP
VIEW
EXIT
```

Example: **`list ?`**

```
ALL
GROUPS
PIN
STATUS
SUBSYSTEM
```

Add

Use the **add** command to add an individual event to an existing group or to create a new group. Group names must start with a letter and are case sensitive. You cannot append an entire subsystem to a group.

Syntax: add *group_name subsystem.event_number*

Example: **`add MyGroup gw.019`**

Note: If the specified group does not exist, the following prompt asks you to confirm the creation of a new group:

```
Group not found. Create new group? (yes or no)
```

Clear

Use the **clear** command to clear all of the ELS configuration information.

Syntax: clear

Example: **`clear`**

```
You are about to clear all ELS configuration information
Are you sure you want to do this (Yes or No):
```

Default

Resets the display or trap setting of an event, group, or subsystem back to a disabled state.

Syntax: default display
trap

display *event* OR *group* OR *subsystem*

Controls the output of the display of messages to the console.

Example: **`default display event snmp.016`**

trap *event* OR *group* OR *subsystem*

Controls the generation of traps to the network management station.

Example: **`default trap subsystem ip`**

Delete

Use the **delete** command to delete an event number from an existing group or to delete the entire group. If the specified event is the last event to be deleted in a group, you will be notified. If *all* is specified instead of *subsystem.event_number*, a prompt asks you to confirm the deletion of the entire group.

Syntax: delete *group_name subsystem.event_number*

Example: **`delete groupa gw.019`**

Display

Use the **display** command to enable message displaying on the console monitor for specific events, a range of events for a subsystem, groups, or subsystems.

Syntax: display event . . .
| range . . .
group . . .
subsystem . . .

event *subsystem.event#*

Displays messages of the specified event (*subsystem.event#*).

Example: **`display event gw.019`**

| range *subsystemname first_event_number last_event_number*

| Where first_event_number is the number of the first event in the specified
| event range, and last_event_number is the number of the last event in the
| specified event range.

| Displays a range of messages for the specified subsystem.

| **Example:** **`display range gw 19 22`**

| Displays events gw.19, gw.20, gw.21, and gw.22.

group *groupname*

Displays messages of a specified group (*groupname*).

Example: **`display group groupb`**

subsystem *subsystemname*

Displays messages associated with the specified subsystem. The following is a list of subsystems that are supported on the router. To find out which subsystems are on your router, type **list subsystems**.

Note: Although ELS supports all of these subsystems, not all devices support all subsystems. See *ELS Messages* for the most current list of supported subsystems.

Subsystem	Description
AI	Auto-device Install
All	All subsystems

Note: Do not display all subsystems for extended periods of time when the router is forwarding live protocol traffic because this causes the router to spend an excessive amount of time communicating with the console. Never display all subsystems when you are communicating with the router via a remote console. This causes the router to spend most of its time communicating with the remote console.

AP2	AppleTalk Phase 2
ARP	Address Resolution Protocol
APPN	Advanced Peer-to-Peer Networking
ATM	Asynchronous Transfer Mode
BAN	Boundary Access Node
BGP	Border Gateway Protocol
BR	Bridging/Routing

BRS	Bandwidth Reservation
BTP	BOOTP relay agent
CLNP	ISO 8473 - CLNP
COMP	Data Compression
DLS	Data Link Switching
DN	DECnet
DOUT	DIALs Server Dial-Out
DNAV	DNA Phase V
DVM	DVMRP Multicast Routing Protocol
ENCR	Data Encryption
ESIS	ISO 9542 - ESIS Protocol
ETH	Ethernet handler
EZ	EasyStart
FLT	Filter library
FRL	Frame Relay
GW	Router base and network library
ICMP	Internet Control Message Protocol
ILMI	Interim Local Management Interface
IP	Internet Protocol
IPPN	IP Protocol Net
IPX	Internetwork Packet Exchange Protocol
ISDN	Integrated-services Digital Network
ISIS	ISO 10589 - ISIS Protocol
ILMI	ATM Interim Local Management Interface
LCS	Logical Channel Station
LEC	ATM LAN Emulation Client
LECS	LAN Emulation Configuration Server
LES	LAN Emulation Server
LLC	Logical Link Control
LSA	Link Services Architecture
LNM	LAN Network Manager
MCF	MAC Filtering
MPC	Multi-Path Channel
MSPF	OSPF Multicast extensions
NBS	NetBIOS Support Subsystem
NOT	Non-supported Protocol Forwarder
OSPF	Open SPF-based Routing Protocol
PPP	Point-to-Point Protocol
RIP	IP Routing Information Protocol
R2MP	AppleTalk Phase 2 Routing Table Management Protocol
SAAL	Signaling ATM Adaptation Layer
SDLC	IBM SDLC
SL	Serial Line Handler
SNMP	Simple Network Management Protocol
SRLY	SDLC Relay
SRT	Source Routing Transparent Bridge
STP	Spanning Tree Protocol
SVC	Switched Virtual Connection
TCP	Transport Control Protocol
TFTP	Trivial File Transfer Protocol
TKR	Token Ring Handler
UDP	User Datagram Protocol
VIN	Banyan VINES
V25B	CCITT/ITU V.25bis

WRS	WAN Restoral/Reroute
XN	XNS/IPX/DDS common processing
XNS	Xerox Networking Systems Protocol
X25	X.25 Protocols
X251	X.25 Physical Layer
X252	X.25 Frame Layer
X253	X.25 Packet Layer
XTP	X.25 Transport Protocol
ZIP2	AppleTalk Phase 2 Zone Information Protocol

Example: **`display subsystem tkr`**

List

Use the **list** command to get updated information regarding ELS settings and listings of selected messages.

Syntax: list
- all
- groups
- pin
- status
- subsystem
- subsystem . . .
- subsystems all
- trace-status

all

Lists information from all the **list** categories.

Example: **`list all`**

groups

Lists the user-defined group names and contents.

Example: **`list groups`**

```
Group: test
GW.019
```

pin

Lists the current number of ELS event messages sent in SNMP traps (per second).

Example: **`list pin`**

```
Pin: 100 events/second
```

status

Lists the subsystems, groups, and events that have been modified by the **display**, **nodisplay**, **trap**, and **notrap** commands.

Example: **`list status`**

subsystem

Lists names, events, and descriptions of all subsystems.

Example: **`list subsystem`**

(Example output from a **list subsystem** command can be found beginning on page 9-9.)

subsystem *subsystem*

Lists all events in a specified subsystem.

Example: **`list subsystem gw`**

```
Event      Level      Message

GW.001     ALWAYS   Copyright 1984 Mass Institute of Technology
GW.002     ALWAYS   Portable CGW %s Rel %s strtd
GW.003     ALWAYS   Unus pkt len %d nt %d int %s/%d
GW.004     ALWAYS   Sys %s q adv alloc %d excd %d
GW.005     ALWAYS   Bffrs: %d avail %d idle   fair %d low %d
GW.006     C-INFO    Pkt frm nt %d int %s/%d for uninit prt, disc
GW.007     C-INFO    Ip err %x nt %d int %s/%d
GW.008     U-INFO    Ip ovfl nt %d int %s/%d, disc
GW.009     UI-ERROR  Nt dwn ip rstrt nt %d int %s/%d
GW.010     UI-ERROR  Ip q len %d no ip buf nt %d int %s/%d
GW.011     U-INFO    Op err %x hst %wo nt %d int %s/%d
GW.012     U-INFO    Op err cnt excd hst %wo nt %d int %s/%d
GW.013     U-INFO    Rtrns cnt excd hst %wo nt %d int %s/%d
GW.014     UI-ERROR  Nt dwn op rstrt nt %d int %s/%d
GW.015     UI-ERROR  Nt dwn to hst %wo nt %d int %s/%d
GW.016     U-INFO    Op ovfl to hst %wo nt %d int %s/%d
GW.017     UE-ERROR  Intfc hdw mssng nt %d int %s/%d
GW.018     U-TRACE   Strt nt slf tst nt %d int %s/%d
GW.019     C-INFO    Slf tst nt %d int %s/%d
GW.020     U-TRACE   Nt pss slf tst nt %d int %s/%d
GW.021     UE-ERROR  Nt up nt %d int %s/%d
GW.022     U-TRACE   Nt fld slf tst nt %d int %s/%d
```

subsystems all
: Lists all events in all subsystems.

  Example: **`list subsystems all`**

trace-status
: Displays information on the status of packet tracing, including configuration and run-time information.

  Example: **`list trace-status`**

```
------------------------ Configuration -----------------------------
Trace Status:ON  Wrap Mode:ON  Decode Packets:ON  HD Shadowing:ON
RAM Trace Buffer Size:100000  Maximum Trace Buffer File Size:10000000
Max Packet Bytes Trace:256  Default Packet Bytes Traced:100
Trace File Record Size:2048  Stop Trace Event: TCP.013
Maximum Hours to HD Shadow:  1
```

Nodisplay

Use the **nodisplay** command to select and turn off messages displaying on the console monitor.

Syntax: nodisplay event. . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*
: Suppresses the displaying of a specified event (*subsystem.event#*).

  Example: **`nodisplay event gw.019`**

range *subsystemname first_event_number last_event_number*

: Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

: Suppresses the displaying of a range of messages for the specified subsystem.

: **Example:** **`nodisplay range gw 19 22`**

Suppresses the display of events gw.19, gw.20, gw.21, and gw.22.

group *groupname*
Suppresses the displaying of messages that were previously added to the specified group (*groupname*).

Example: **`nodisplay group groupb`**

subsystem *subsystemname*
Suppresses the displaying of messages associated with the specified subsystem.

Example: **`nodisplay subsystem tkr`**

Notrace

Disables packet trace for the specified event/range/subsystem/group.

Syntax: notrace event . . .
range . . .
group . . .
subsystem . . .

event subsystem.*event#*
Suppresses the sending of packet trace data for the specified event#

Example: **`notrace event atm.088`**

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Disables the sending of packet trace data for a range of messages for the specified subsystem.

Example: **`notrace range gw 19 22`**

Suppresses the sending of packet trace data for events gw.19, gw.20, gw.21, and gw.22.

group *groupname*
Suppresses the sending of packet trace data that was previously added to the specified group (groupname).

Example: **`notrace group groupb`**

subsystem *subsystemname*
Suppresses the sending of packet trace data for the specified subsystem (subsystemname).

Example: **`notrace subsystem atm`**

Notrap

Use the **notrap** command to select and turn off messages so that they are no longer sent to a network management workstation in SNMP traps.

Syntax: notrap event . . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*

Suppresses the sending of the specified message in an SNMP trap (*subsystem.event#*).

Example: **`notrap event gw.019`**

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Suppresses the sending of messages for the events in the specified range for the specified subsystem in SNMP traps.

Example: **`notrap range gw 19 22`**

Suppresses the sending of messages for events gw.19, gw.20, gw.21, and gw.22 in SNMP traps.

group *groupname*

Suppresses the sending of messages in SNMP traps that were previously added to the specified group (*groupname*).

Example: **`notrap group groupb`**

subsystem *subsystemname*

Suppresses the sending of messages in SNMP traps that are associated with the specified subsystem.

Example: **`notrap subsystem tkr error`**

Set

Use the **set** command to set the maximum number of traps per second, to set the timestamp feature, or to set tracing options for ATM devices.

Syntax: set pin . . .
timestamp . . .
trace . . .

pin *max_traps*

Use the **set pin** command to set the pin parameter to the maximum number of traps that can be sent on a per-second basis. Internally, the pin resets every tenth of a second. (One tenth of the number (*max_traps*) is sent every tenth of a second.)

Example: **`set pin 100`**

timestamp timeofday OR uptime OR off

Allows you to turn on message timestamping so that either the time of day or uptime (number of hours, minutes, and seconds, but no date, since the router was last initialized) appears next to each message. Set timestamp can also be turned off.

Use the **set timestamp** command to enable one of the following timestamp options.

Example: **`set timestamp  timeofday`**

timeofday

Adds an HH:MM:SS prefix to each ELS message indicating the time of the occurrence during a 24-hour day.

uptime

Adds an HH:MM:SS prefix to each ELS message indicating the time of the occurrence during a 100-hour cycle. After 100 hours of uptime, the uptime counter returns to zero to begin another 100-hour cycle.

off

Turns off the ELS timestamp prefix.

trace

Use the **set trace** command to configure tracing options for ATM devices. When tracing options are configured from the monitoring console, the changes take effect immediately, and return to their previously configured settings when the device is rebooted.

Note: Tracing should be used only under the direction of trained support personnel. Tracing, especially when used with disk-shadowing enabled, uses device resources and can impact overall performance and throughput.

Syntax: set trace
- decode
- default-bytes-per-port
- max-bytes-per-port
- off
- on
- reset
- wrap-mode

decode *off/on*

Turns packet decoding on or off. Packet decoding is not supported by all components.

default-bytes-per-pkt *bytes*

Sets the default number of bytes traced. This value is used if a value is not specified by the component doing the tracing.

max-bytes-per-pkt *bytes*

Sets the maximum number of bytes traced for each packet.

wrap-mode *off/on*

Turns the trace buffer wrap mode on or off. If wrap mode is on and the trace buffer is full, previous trace records will be overwritten by new trace records as necessary to continue tracing.

Example 1: `set trace decode on`

Example 2: `set trace default-bytes-per-packet 64`

Example 3: `set trace off`

Trace

Enables packet trace for the specified event/range/subsystem/group. When the **trace** command is used from the `ELS Config>` prompt, the changes become part of the configuration, and a reboot is required to activate the changes.

Syntax: trace
- event . . .
- range . . .
- group . . .
- subsystem . . .

event *subsystem.event#*

Causes the specified trace event (*subsystem.event#*) to be displayed on the system console.

Example: `trace event gw.019`

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Causes the trace events in the specified range for the specified subsystem to be displayed on the system console.

Example: `trace range gw 19 22`

Causes the trace events gw.19, gw.20, gw.21, and gw.22 to be displayed on the system console.

group *groupname*

Allows trace events that were previously added to the specified group to be displayed on the router console.

Example: `trace group groupb`

subsystem *subsystemname*

Allows trace events associated with the specified subsystem to be displayed on the router console.

Example: `trace subsystem gw`

Trap

Use the **trap** command to select the message to be sent to the remote SNMP network management workstation. A remote SNMP network management workstation is an IP host in the network acting as an SNMP manager.

Syntax: trap event . . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*

Causes the specified message (*subsystem.event#*) to be sent to a network management workstation in an SNMP trap.

Example: `trap event gw.019`

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Causes the messages that are in the specified range for the specified subsystem to be sent to a network management workstation in an SNMP trap.

Example: `trap range gw 19 22`

Causes the messages in events gw.19, gw.20, gw.21, and gw.22 to be sent to a network management workstation in an SNMP trap.

group *groupname*

Allows messages that were previously added to the specified group to be sent to a network management workstation in an SNMP trap.

Example: **`trap group groupb`**

subsystem *subsystemname*

Allows messages associated with the specified subsystem to be sent to a management station in an SNMP trap.

Example: **`trap subsystem gw`**

Note: Messages for the IP, ICMP, ARP and UDP subsystems cannot be sent in SNMP traps because these areas are or may be used in the process of sending the SNMP trap. This could lead to an infinite loop of traffic putting an undue strain on the router.

Exit

Use the **exit** command to return to the CONFIG prompt.

Syntax: exit

Example: **`exit`**

Chapter 9. Monitoring the Event Logging System (ELS)

This chapter describes how to monitor events logged by ELS and how to use the ELS console commands. The information includes the following sections:

If you need more information on the Event Logging System and how to interpret ELS event messages, refer to Chapter 8, "Using and Configuring the Event Logging System (ELS)" on page 8-1.

Using ELS

To use ELS effectively, it is recommended you take the following steps:

- Know what you want to see before using the ELS system. Clearly define the problem or events that you want to see before using the MONITR process.
- Execute the command **nodisplay subsystem all all** to turn off all ELS messages.
- Turn on only those messages that relate to the problem you are experiencing.
- Use the *IBM Nways Event Logging System Messages Guide* to determine which messages you are seeing are normal.

When you initially view ELS from the MONITR process, you will see a considerable amount of information. Because the router cannot buffer and display every packet under moderate to heavy loads the buffers are flushed. When this occurs the following message is displayed:

```
xx messages flushed
```

The router does not save these messages. When this message appears, you may want to tailor the ELS output to display only that information that is important to the current task you are monitoring.

Managing ELS Message Rotation

It is also important to note that the ELS messages continually rotate through the router's buffers. To stop and restart the displaying of ELS messages, use the following key combinations:

Ctrl S	to pause scrolling
Ctrl Q	to resume scrolling
Ctrl P	to go back to the last process

You may also want to capture the ELS output to a file. You can do this by starting a script file or log file from your location when Telneting to a router. You can also do this by attaching a PC to the router's console port and starting a log file from within the terminal emulation package. This information is needed to help Customer Service diagnose a problem.

Capturing ELS Output Using a Telnet Connection on a UNIX Host

You can use a Telnet connection on an AIX or UNIX host to capture the ELS messages on your screen to a file on the host. Before beginning, make sure you have set up ELS for the messages you want to capture by using the ELS console commands in this chapter.

To capture the ELS output to a file on an AIX or UNIX host, follow these steps:

1. From the host, enter **telnet** *router_ip_addr* **| tee** *local_file_name*

   router_ip_addr is the IP address of the router
   local_file_name is the name of the file on the host where you want the ELS messages to be saved.
   The **tee** command displays the ELS messages on your screen and, at the same time, copies them to the local file.

2. From the OPCON prompt (*****), enter **t 2**. This accesses the MONITR process, which is the process that displays ELS messages on your screen. Depending on which ELS messages you configured, you should see ELS messages appearing on the screen.

   As long as you are in the MONITR process, all ELS messages will be written to the local file. When you exit the MONITR process (by entering **Ctrl** **P**) or terminate the Telnet session, the logging of messages to the local file will stop.

Configuring ELS So Event Messages Are Sent In SNMP Traps

ELS can be configured so that event messages are sent to a network management workstation in an SNMP enterprise-specific trap. These traps are useful for reporting status and diagnostic results, and are often used for remote monitoring of a 2210. When ELS is configured appropriately, an SNMP trap will be generated each time the selected event occurs. For more information about SNMP, see *Protocol Configuration and Monitoring Reference*.

To tell ELS that a specific event should be activated to be sent as an SNMP trap, at the `ELS config>` prompt or at the `ELS>` prompt, using IP as an example, type:

```
trap event ip.007
```

Note: If you are at the `ELS config>` prompt, you will need to reboot.

To enable the ELS enterprise-specific trap, follow these steps:

1. At the `SNMP config>` prompt, using **public** as an example, type:

   ```
   SNMP config> add address public <network manager IP address>

   SNMP config> enable trap enterprise public

   SNMP config> set community access read_trap public
   ```

   Note: You will need to reboot to activate these changes.

2. Enable your network management station to receive and properly display the enterprise-specific traps.

You can follow the steps above for trapping groups, subsystems, and events.

Using ELS to Troubleshoot a Problem

Events occur continuously while the router is operating. They can be caused by any of the following reasons:

- System activity
- Status changes
- Service requests
- Data transmission and reception
- Data and internal errors

When an event occurs, ELS receives data from the system that identifies the source and nature of the event. Then ELS generates a message that uses the data received as part of the message.

When trying to troubleshoot a particular problem, display those messages that relate to the problem. For example, if you are experiencing a problem with bridging, turn on the bridging messages:

display subsystem srt all
display subsystem br all

Initially, because of the rapid pace of messages scrolling across the screen, you may want to record the numbers you see and look those up in the manual. Once you become familiar with different types of messages being displayed for a particular protocol, you can turn on and turn off only those messages that contain the information that you require to troubleshoot a problem. The following sections list specific ELS examples. Keep in mind that different problems may require different steps.

ELS Example 1

You are interested in looking at the frequency of polling on a Token-Ring interface, and finding out whether the polls are successful.

```
ELS> nodisplay subsystem all all

ELS> display subsystem tkr all

Ctrl  P

* t 2
```

As the messages begin to scroll by, look for ELS message tkr.031.

ELS Example 2

SRTB bridging is not working.

1. Check the configuration.
2. Use the GWCON bridging console to verify that the bridging interfaces are enabled.
3. Enter:

```
* t 6

config> event

ELS config> nodisplay subsystem all all

ELS config> display subsystem srt all

ELS config> exit

config> Ctrl P
```

4. Restart the routing subsystem. When the subsystem has restarted, enter the following:

```
* t 2
```

As the messages begin to scroll by, look for messages srt.071 through srt.075. If you see one of these messages, you are not licensed to use one or more of the bridging features.

ELS Example 3

Router cannot communicate with an IPX server on an Ethernet.

1. At the OPCON prompt, enter the **status** command to find the PID (process ID) of GWCON. (See step 1 on page 8-3 in "ELS Configuration Environment" on page 8-3 for a sample output of the **status** command.)

```
* status
```

2. Enter the **talk** command and the PID for GWCON.

```
* talk 5
```

   The console displays the GWCON prompt (+). If the prompt does not appear when you first enter GWCON, press **Return**.

3. At the GWCON prompt (+), enter **IPX** to access the IPX console prompt (IPX>).
4. At the IPX console prompt, enter the **slist** command to verify that the server is listed. (See the section on monitoring IPX in the *Protocol Configuration and Monitoring Reference* for information on the slist command.)
5. Check the IPX configuration.
6. Enter the following:

```
* t 5

+ event

ELS> nodisplay subsystem all all

ELS> display subsystem IPX all

ELS> display subsystem eth all

ELS> Ctrl P

* t 2
```

As the messages begin to scroll by, look for ELS message eth.006. This indicates that the server has a bad econfig.

Entering and Exiting the ELS Console Environment

The ELS console environment (available from the GWCON process) is characterized by the ELS> prompt. Commands entered at this prompt modify the current ELS parameter settings. These commands are described Chapter 9, “Monitoring the Event Logging System (ELS)” on page 9-1.

To enter the ELS console environment from OPCON:

1. At the OPCON prompt, enter the **status** command to find the PID (process ID) of GWCON. (See step 1 on page 8-3 in “ELS Configuration Environment” on page 8-3 for a sample output of the **status** command.)

   ```
   * status
   ```

2. Enter the **talk** command and the PID for GWCON.

   ```
   * talk 5
   ```

   The console displays the GWCON prompt (+). If the prompt does not appear when you first enter GWCON, press **Return**.

3. At the GWCON prompt, enter the following command to access ELS:

   ```
   + event
   ```

   The console displays the ELS console prompt (ELS>). Now, you can enter ELS console commands.

To leave the ELS console environment, enter the **exit** command.

ELS Console Commands

This section summarizes and then explains all the ELS console commands. After accessing the ELS Console environment, you can enter ELS console commands at the ELS> prompt.

Table 9-1. ELS Console Command Summary

Command	Function
? (Help)	Lists the ELS console commands or lists the options associated with specific commands.
Clear	Clears messages associated with specific events, groups, or subsystems.
Display	Enables message display on the console.
Exit	Exits the ELS console process and returns the user to GWCON.
List	Lists information on ELS settings and messages.
Nodisplay	Disables message display on the console.
Notrace	Disables trace event display on the console.
Notrap	Keeps messages from being sent out in SNMP traps to the network management workstation.
Packet-trace	Provides an enhanced central environment for setting and listing active packet tracing parameters.
Remove	Frees up memory by erasing stored information.
Restore	Clears current settings and reloads initial ELS configuration.
Retrieve	Reloads the saved ELS configuration.
Save	Stores the current configuration.
Set	Sets the pin parameter and the timestamp feature.
Statistics	Displays available subsystems and pertinent statistics.
Trace	Enables trace event display on the console.
Trap	Allows messages to be sent to a network management workstation in SNMP traps.
View	Allows viewing of traced packets.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
CLEAR
DISPLAY
LIST
NODISPLAY
NOTRACE
NOTRAP
PACKET-TRACE
REMOVE saved state
RESTORE initial state
RETRIEVE saved state
SAVE current state
SET
STATISTICS
TRACE
TRAP
VIEW
EXIT
```

Example: **`list ?`**

```
ALL
ACTIVE
EVENT
GROUPS
PIN
SUBSYSTEMS
TRACE
```

Clear

Use the clear command to disable both the display and trap commands as they relate to specific events, groups, or subsystems.

Syntax: clear event . . .
group . . .
subsystem . . .

event *subsystem. event#*

Disables the displaying or trapping of messages for the specified event (*subsystem.event#*).

Example: **`clear event gw.019`**

group *group.name*

Disables the displaying or trapping of messages for the specified group (*group.name*). specified group (*groupname*).

Example: **`clear group groupb`**

subsystem *subsystem.name*

Disables the displaying or trapping of messages associated with the specified subsystem (*logging level*). If you do not specify a logging level, all messages for that subsystem are disabled.

Example: **`clear subsystem gw`**

Display

Use the display command to enable the message display on the console monitor for specific events.

Syntax: display event . . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*

Displays messages for the specified event (subsystem.event#).

Example: **`display event gw.019`**

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event in the specified event range.

Displays a range of messages for the specified subsystem.

Example: **`display range gw 19 22`**

Displays events gw.19, gw.20, gw.21, and gw.22.

group *groupname*
Displays messages of a specified group (*groupname*).

Example: **`display group groupb`**

subsystem *subsystem.name*
Displays any messages associated with the specified subsystem (*logging level*). If you do not specify a logging level, all messages for that subsystem are turned on.

Example: **`display subsystem tkr`**

List

Use the list command to get updated information regarding ELS settings and to get listings of selected messages.

Syntax: list
all
active . . .
event . . .
groups . . .
pin
subsystems . . .
trace-status

all
Lists all subsystems, defined groups, enabled subsystems, enabled events, and pins.

Example: **`list all`**

active *subsystem.name*
Displays the events that are active for a specific subsystem and the count of the occurrence of the messages.

Example: **`list active ip`**

```
EventActiveCount
IP.00789354
ETH.009D10
Subsystem X25: no event active
```

event *subsystem.event#*
Displays the logging level, the message, and the count of the specified event.

Example: **`list event ip.007`**

```
Level: p-TRACE
Message: source_ip_address -> destination_ip_address
Active:  Count: 84182
```

groups *group.name*
Displays the user-defined group names.

Example: **`list groups`**

pin
Lists the current number of ELS event messages sent per second in SNMP traps. This is a threshold value that can be used to reduce the amount of SNMP trap traffic.

Example: **`list pin`**

```
Pin: 100 events/second
```

subsystem *subsystem.name*

Lists event names, the total number of events that have occurred, and their descriptions.

Note: Although ELS supports all of these subsystems, not all devices support all subsystems. See *ELS Messages* for the most current list of supported subsystems.

Example: `list subsystem`

Name	Events	Description
ALL		All subsystems
GW	101	Router base and network library
FLT	7	Filter Library
BRS	5	Bandwidth Reservation
ARP	142	Address Resolution Protocol
IP	100	Internet Protocol
ICMP	21	Internet Control Message Protocol
TCP	57	TCP
UDP	6	User Datagram Protocol
BTP	13	BOOTP relay agent
RIP	22	IP Routing Information Protocol
OSPF	73	Open SPF-Based Routing Protocol
MSPF	17	OSPF Multicast extensions
TFTP	29	TFTP Protocol
SNMP	28	Simple Network Management Protocol
DVM	21	DVMRP Multicast Routing Protocol
DN	115	DECnet
XN	21	XNS/IPX/DDS common processing
IPX	110	Internetwork Packet Exchange Protocol
CLNP	58	ISO 8473 - CLNP
ESIS	24	ISO 9542 - ESIS Protocol
ISIS	58	ISO 10589 - ISIS Protocol
DNAV	26	DNA Phase V
AP2	70	AppleTalk Phase 2
ZIP2	51	Appletalk Phase 2 Zone Information Protocol
R2MP	38	Appletalk Phase 2 Routing Table Management Protocol
VIN	79	Banyan VINES
SRT	94	Source Routing Transparent Bridge
STP	32	Spanning Tree Protocol
BR	30	Bridge/Routing
SRLY	28	SDLC Relay
ETH	47	Ethernet Handler
SL	35	Serial Line Handler
TKR	45	Token Ring Handler
X25	53	X.25 Protocols
FDDI	27	FDDI Handler
SDLC	95	IBM SDLC
FRL	97	Frame Relay
PPP	186	Point-to-Point
X251	16	X.25-Physical-Layer
X252	34	X.25-Frame-Layer
X253	42	X.25-Packet-Layer
ISDN	43	Integrated Services Digital Network
IPPN	4	IP Protocol Net
WRS	33	WAN Restoral
LNM	60	LNM
LLC	168	Logical Link Control
BGP	74	Border Gateway Protocol
MCF	9	MAC Filtering
DLS	497	Data Link Switching
V25B	28	CCITT/ITU V.25bis
BAN	29	Boundary Access Node
COMP	26	Data Compression Engines
NBS	50	NetBIOS Support Subsystem
ATM	216	Asynchronous Transfer Mode
LEC	174	ATM LAN Emulation Client
APPN	28	Advanced Peer-to-Peer Networking
ILMI	23	ATM Interim Local Management Interface
SAAL	26	ATM Signalling ATM Adaption Layer
SVC	26	ATM Signalling
LES	361	LAN Emulation Services
LECS	145	LAN Emulation Configuration Server
EVLOG	1	EventLog() error logging system
NOT	15	Forwarder messages not loaded
NHRP	211	Next Hop Resolution Protocol
XTP	58	X.25 Transport

```
LCS        22     LCS Handler
LSA        61     LSA Handler
MPC        30     MPC Handler
SCSP       34     Server Cache Synchronization Protocol
ALLC       36     ATM LLC (RFC1483)
NDR        38     Network Dispatcher Router Feature
MLP        93     Multilink-PPP
SEC        30     Security Protocols
ENCR        4     Data Encryption Engines
PM          6     Presence Manager
DGW         9     Default Gateway
QLLC       54     QLLC-Packet-LayerName  Events  Description
VLAN       20     Virtual LAN
```

subsystem *subsystem.name*

Lists all events, logging levels, and messages for the specified subsystem.

Example: **`list subsystem eth`**

```
Event      Level      Message
ETH.001    P-TRACE    brd rcv unkwn type packet_type source_Ethernet_address ->
                      destination_Ethernet_address nt network
ETH.002    UE-ERROR   rcv unkwn typ packet_type source_Ethernet_address ->
                      destination_Ethernet_address nt network
ETH.010    C-INFO     LLC unk SAP DSAP source_Ethernet_address ->
                      destination_Ethernet_address nt network
```

subsystems all

Lists all events, logging levels, and messages for every event that has occurred on the router.

Example: **`list subsystems all`**

trace-status

Displays information on the status of ATM packet tracing, including configuration and run-time information.

Example: **`list trace-status`**

```
------------------------ Configuration -----------------------------
Trace Status:ON  Wrap Mode:ON  Decode Packets:OFF  HD Shadowing:OFF
RAM Trace Buffer Size:100000  Maximum Trace Buffer File Size:10000000
Default Packet Bytes Traced:100  Max Packet Bytes Traced:256
----------------------- Run-time Status ---------------------------
Packets in RAM Trace Buffer:535   Free Trace Buffer Memory:180
Trace Errors:22  First Packet:23  Last Packet:557
Trace Buffers Shadowed to HD:0  Trace Buffer File Size:0
```

- “Trace Status” in the LIST TRACE-STATUS display will indicate OFF when STOP-ON-EVENT action occurs.
- “HD Shadowing” in the LIST TRACE-STATUS display will indicate OFF when STOP-ON-EVENT action occurs or when Time Limit is exceeded.
- “Trace Buffer File Size” will display “<wrapped>” when a wraparound has occurred in the trace file.
- If disk-shadowing time limit is exceeded, but there has not been a trace record written since the time expired, then“HD-Shadowing Time Exceeded? NO <Next trace will turn it OFF>” will be displayed. When the next trace record has been written, then “HD-Shadowing Time Exceeded? YES” will be displayed.

`ELS Config>`**LIST TRACE** command under Talk-6 displays information similar to the following:

```
------------------------ Configuration -----------------------------
Trace Status:ON  Wrap Mode:ON  Decode Packets:ON  HD Shadowing:ON
RAM Trace Buffer Size:100000  Maximum Trace Buffer File Size:10000000
Max Packet Bytes Trace:256  Default Packet Bytes Traced:100
Trace File Record Size:2048  Stop Trace Event: TCP.013
Maximum Hours to HD Shadow:  1
```

Nodisplay

Use the **nodisplay** command to select and turn off messages displaying on the console monitor.

Syntax: nodisplay event . . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*
Suppresses the displaying of messages for the specified event.

Example: **`nodisplay event gw.019`**

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Suppresses the displaying of a range of messages for the specified subsystem.

Example: **`nodisplay range gw 19 22`**

Suppresses the display of events gw.19, gw.20, gw.21, and gw.22.

group *group.name*
Suppresses the displaying of messages that were previously added to the specified group (*group.name*).

Example: **`nodisplay group groupb`**

subsystem *subsystem.name*
Suppresses the displaying of messages associated with the specified subsystem (*logging level*).

Example: **`nodisplay subsystem tkr`**

Notrace

Use the **notrace** command to stop display of selected trace events at the console.

Syntax: notrace event . . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*
Suppresses the display of the specified tracing event.

Example: **`notrace event gw.019`**

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Disables the sending of packet trace data for a range of messages for the specified subsystem.

Example: **notrace range gw 19 22**

Suppresses the sending of packet trace data for events gw.19, gw.20, gw.21, and gw.22.

group *groupname*

Suppresses the display of tracing events related to the specified group (*groupname*).

Example: **notrace group groupb**

subsystem *subsystemname*

Suppresses the display of tracing events that are associated with the specified subsystem.

Example: **notrace subsystem atm error**

notrace subsystem atm

Notrap

Use the **notrap** command to select and turn off messages so that they are no longer sent to a network management workstation in SNMP traps.

Syntax: notrap event. . .
range . . .
group . . .
subsystem . . .

event *subsystem.event#*

Suppresses the sending of the specified message in an SNMP trap (*subsystem.event#*).

Example: **notrap event gw.019**

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Suppresses the sending of messages for the events in the specified range for the specified subsystem in SNMP traps.

Example: **notrap range gw 19 22**

Suppresses the sending of messages for events gw.19, gw.20, gw.21, and gw.22 in SNMP traps.

group *groupname*

Suppresses the sending of messages in SNMP traps that were previously added to the specified group (*groupname*).

Example: **notrap group groupb**

subsystem *subsystemname*

Suppresses the sending of messages in SNMP traps that are associated with the specified subsystem.

Example: `notrap subsystem tkr error`

Packet Trace

Use the **packet-trace** command to display/enable/disable packet tracing information for various subsystems. To enter the Packet-Trace Console from the ELS Console, type **packet-trace** under the ELS> prompt:

```
ELS>packet-trace
Packet Trace Console
ELS Packet Trace>
```

Use the **Exit** command when you are finished using Packet Trace.

For complete command descriptions, see “Packet-trace Console Commands” on page 9-20

Remove

Use the **remove** command to free up memory by erasing stored information. If you have previously saved the current configuration with the **save** command, remove allows you to erase the saved configuration.

Syntax: remove

Example: `remove`

Restore

Use the **restore** command to clear all current settings (except counters) and reload the initial ELS configuration. To retain the current settings, use the **save** command before restoring the initial configuration.

Syntax: restore

Example: `restore`

Retrieve

Use the **retrieve** command to reload the saved ELS configuration. If you have previously saved the current configuration with the **save** command, use **retrieve** to reload it. **Retrieve** does not erase the saved configuration after it executes. To erase the saved configuration, use the **remove** command.

Syntax: retrieve

Example: `retrieve`

Save

Use the **save** command to store the current configuration (except counters). **Save** does not affect the default configuration (the one you set with the configuration commands). Use **save** after modifying the configuration with the console commands with the intention of saving this configuration over a restart. There can be only one saved configuration at a time. To reload the saved configuration, use the **retrieve** command.

Syntax: save

Example: `save`

Set

Use the **set** command to set the maximum number of traps per second, to set the timestamp feature, or to set the tracing options.

pin

Use the **set pin** command to set the pin parameter to the maximum number of traps that can be sent on a per-second basis. Internally, the pin resets every tenth of a second. (One tenth of the number *max_traps* is sent every tenth of a second.)

Syntax: set *pin max_traps*

Example: `set pin 100`

timestamp

Allows you to turn on message timestamping so that either the time of day or uptime (number of hours, minutes, and seconds, but no date, since the router was last initialized) appears next to each message, or to turn off message timestamping.

Note: If you turn on timestamping, you must remember to go back into the CONFIG process and set the router's date and time using the time command. Otherwise, all messages will come out with 00:00:00, or negative numbers in the hours, minutes, and/or seconds, for example 00:-4:-5.

Use the **set timestamp** command to enable one of the following timestamp options:

timeofday Adds an HH:MM:SS prefix to each ELS message indicating the time of the occurrence during a 24-hour day.

uptime Adds an HH:MM:SS prefix to each ELS message indicating the time of the occurrence during a 100-hour cycle of uptime for the router. After 100 hours of uptime, the uptime counter returns to zero to begin another 100-hour cycle.

off Turns off the ELS timestamp prefix.

Syntax: set timestamp timeofday OR uptime OR off

Example: `set timestamp timeofday`

trace

Use the **set trace** command to configure tracing options. When tracing options are configured from the monitoring console, the changes take effect immediately, and return to their previously configured settings when the device is rebooted.

Syntax: set trace
decode . . .
default-bytes-per-port . . .
max-bytes-per-port . . .
off
on
reset
wrap-mode . . .

decode *off / on*
Turns packet decoding on or off. Packet decoding is not supported by all components.

default-bytes-per-pkt *bytes*
Sets the default number of bytes traced. This value is used if a value is not specified by the component doing the tracing.

max-bytes-per-pkt *bytes*
Sets the maximum number of bytes traced for each packet.

off
Disables packet tracing.

on
Enables packet tracing.

reset
Clears the trace buffer and resets all associated counters.

wrap-mode *off/on*
Turns the trace buffer wrap mode on or off. When wrap mode is enabled and the trace buffer is full, previous trace records will be overwritten by new trace records as necessary to continue tracing.

Example $_1$: **`set trace decode on`**

Example $_2$: **`set trace default-bytes-per-packet 64`**

Example $_3$: **`set trace off`**

Statistics

Use the **statistics** command to display a list of all of the available subsystems and their statistics.

Note: The following example may not match your display exactly. The output of the command depends on the version and release of the installed software.

Syntax: statistics

Example: **`statistics`**

```
Subsys  Vector  Exist   String  Active    Heap

  GW      105     101      3411      0         0
 FLT       20       7       184      0         0
 BRS       50       5       201      0         0
 ARP      150     142      7030      0         0
  IP      100     100      2463      2        20
ICMP       30      21       529      0         0
 TCP       60      57      2420      0         0
 UDP       10       6       179      0         0
 BTP       40      13       695      0         0
 RIP       30      22       474      0         0
OSPF       80      73      2859      0         0
MSPF       40      17       593      0         0
TFTP       35      29       819      0         0
SNMP       30      28       821      0         0
 DVM       30      21       589      0         0
  DN      140     115      5842      0         0
  XN       35      21       780      0         0
 IPX      110     110      4705      0         0
```

CLNP	80	58	1763	0	0
ESIS	40	24	716	0	0
ISIS	80	58	2422	0	0
DNAV	50	26	1314	0	0
AP2	80	70	1755	0	0
ZIP2	60	51	1859	0	0
R2MP	50	38	1185	0	0
VIN	90	79	3159	0	0
SRT	120	94	5040	0	0
STP	60	32	1590	0	0
BR	50	30	1616	0	0
SRLY	30	28	1409	0	0
ETH	60	47	1098	0	0
SL	50	35	584	0	0
TKR	60	45	2031	0	0
X25	70	53	1909	0	0
FDDI	30	27	1155	0	0
SDLC	100	95	4263	0	0
FRL	130	97	6068	0	0
PPP	190	186	6394	0	0
X251	50	16	546	0	0
X252	50	34	996	0	0
X253	50	42	1649	0	0
ISDN	50	43	1994	0	0
IPPN	20	4	132	0	0
WRS	40	33	1938	0	0
LNM	70	60	3137	0	0
LLC	170	168	9840	0	0
BGP	80	74	2477	0	0
MCF	15	9	244	0	0
DLS	500	497	24340	0	0
V25B	30	28	1058	0	0
BAN	30	29	1223	0	0
COMP	80	26	1050	0	0
NBS	100	50	3029	0	0
ATM	300	216	10808	0	0
LEC	200	174	7258	0	0
APPN	100	28	467	0	0
ILMI	150	23	487	0	0
SAAL	30	26	621	0	0
SVC	30	26	465	0	0
LES	400	361	22333	0	0
LECS	150	145	5666	0	0
EVLOG	1	1	105	0	0
NOT	25	15	508	0	0
NHRP	250	211	8193	0	0
XTP	64	58	2271	0	0
ESC	150	67	3122	0	0
LCS	40	22	858	0	0
LSA	70	61	3506	0	0
MPC	130	30	1677	3	44
SCSP	40	34	1234	0	0
ALLC	50	36	1842	0	0
NDR	50	38	1150	0	0
MLP	100	93	4006	0	0
SEC	50	30	688	0	0
ENCR	100	4	194	0	0
PM	25	6	120	0	0

```
 DGW      20       9       238     0         0
QLLC      55      54      2411     0         0

Total   6490    4942    215805     5        64

Maximum:7976 vector, 155 subsystem
Memory:71784/620 vector+ 81256/217714 data+ 64 heap=371438Subsys
```

Subsys	Name of subsystem
Vector	Maximum size of subsystem
Exist	Number of events defined in this subsystem
String	Number of bytes used for message storage in this subsystem
Active	Number of active (displayed, trapped, or counted) events in the subsystem
Heap	Dynamic memory in use by subsystem

Trace

Use the **trace** command to select the trace events to be displayed on the system console.

Syntax: trace event . . .
| range . . .
group . . .
subsystem . . .

event *subsystem.event#*

Causes the specified trace event (*subsystem.event#*) to be displayed on the system console.

Example: **`trace event gw.019`**

| range *subsystemname first_event_number last_event_number*

| Where first_event_number is the number of the first event in the specified | event range, and last_event_number is the number of the last event of the | specified event range.

| Causes the trace events in the specified range for the specified subsystem to | be displayed on the system console.

| **Example:** **`trace range gw 19 22`**

| Causes the trace events gw.19, gw.20, gw.21, and gw.22 to be displayed on | the system console.

group *groupname*

Allows trace events that were previously added to the specified group to be displayed on the router console.

Example: **`trace group groupb`**

subsystem *subsystemname*

Allows trace events associated with the specified subsystem to be displayed on the router console.

Example: **`trace subsystem gw`**

Trap

Use the **trap** command to select the message to be sent to the remote SNMP network management workstation. A remote SNMP network management workstation is an IP host in the network acting as an SNMP manager.

Syntax: trap
- event . . .
- range . . .
- group . . .
- subsystem . . .

event *subsystem.event#*

Causes the specified message (*subsystem.event#*) to be sent to a network management workstation in an SNMP trap.

Example: `trap event gw.019`

range *subsystemname first_event_number last_event_number*

Where first_event_number is the number of the first event in the specified event range, and last_event_number is the number of the last event of the specified event range.

Causes the messages that are in the specified range for the specified subsystem to be sent to a network management workstation in an SNMP trap.

Example: `trap range gw 19 22`

Causes the messages in events gw.19, gw.20, gw.21, and gw.22 to be sent to a network management workstation in an SNMP trap.

group *groupname*

Allows messages that were previously added to the specified group to be sent to a network management workstation in an SNMP trap.

Example: `trap group groupb`

subsystem *subsystemname*

Allows messages associated with the specified subsystem to be sent to a management station in an SNMP trap.

Example: `trap subsystem gw`

Note: Messages for the IP, ICMP, ARP and UDP subsystems cannot be sent in SNMP traps because these areas are or may be used in the process of sending the SNMP trap. This could lead to an infinite loop of traffic putting an undue strain on the router.

View

Use the **view** command to view traced packets.

Syntax: view
- current
- first
- jump
- last
- next
- prev
- search ...

current
Displays the current trace packet. If the current packet is not valid, the first packet in the trace buffer is displayed.

first
Displays the first traced packet in the trace buffer.

jump *n*
Displays the traced packet *n* packets ahead of or behind the current packet.

last
Displays the last traced packet in the trace buffer.

next
Displays the next traced packet.

prev
Displays the previous traced packet.

search *hexstring*
Displays the next traced packet that contains the specified hex string.

Example: view current

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

Packet-trace Console Commands

This section describes the Packet-trace Console commands. After accessing the Packet-trace Console environment, you can enter Packet-trace Console commands at the `ELS Packet Trace>` prompt.

Table 9-2. Packet Trace Console Command Summary

Command	Function
? (Help)	Lists the Packet Trace console commands or lists the options associated with specific commands.
Off	Disables packet tracing.
On	Enables packet tracing. Prompts for memory trace buffer size if not previously set.
Reset	Clears the trace buffer and resets all associated counters.
Set	Configures tracing options.
Subsystems	Activates tracing for the ATM subsystems, or displays a summary.
Trace-status	Displays information on the status of ATM packet tracing, including configuration and run-time.
View	Provides View Captured Packet Trace Buffers Console
Exit	Exits the Packet Trace Console process and returns you to the ELS console.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its specific options.

Syntax: ?

Example: **?**

```
OFF
ON
RESET
SET
SUBSYSTEMS
TRACE-STATUS
VIEW
EXIT
```

Off

Use the **off** command to disable packet tracing.

Syntax: off

Example: `off`

On

Use the **on** command to enable packet tracing.

Syntax: on

Example: `on`

Reset

Use the **reset** command to clear the trace buffer and reset all associated counters.

Syntax: reset

Example: `reset`

Set

Use the **set** command to configure tracing options.

Syntax: Set
- decode
- default-bytes-per-pkt
- disk-shadowing
- max-bytes-per-pkt
- memory-trace-buffer-size
- stop-event
- wrap-mode
- exit

Example: `set memory-trace-buffer-size`

Example: `set decode`

For an explanation of the `set` command, see “Set” on page 9-15.

Subsystems

Use the **subsystems** command to activate tracing for the ATM subsystems or display a summary.

Syntax: subsystems atm
lec
summary

Example: subsystems atm

Example:

```
subsystems atm
Network number? 0
ATM Interface is selected
on  off  list   [list]? on
Note that SVC  uses VPI = 0, VCI = 5
and ILMI uses VPI = 0, VCI = 16
Beginning of VPI range [0]?
End of VPI range [0]?
Beginning of VCI range [0]? 16
End of VCI range [0]? 16
Tracing event ATM.88: ATM frames
```

Example:

```
subsystems lec
Network number? 1
ATM Emulated LAN is selected
on  off  list   [list]? on
Trace which types of frames (data, control, both) [both]?
Tracing event LEC.11: data frames over ATM Forum LEC: interface 1
Tracing event LEC.12: control frames over ATM Forum LEC: interface 1
Note that if the user DISABLEs and TESTs this LEC interface,
the LEC trace settings from Talk 6 Config will take effect.
```

Example:

```
subsystems summary
Subsystems Being Traced

ATM        net number = 0, VPI Range:     0 -     0
                           VCI Range:    16 -    16
LEC        net number = 1
```

Trace-Status

Use the **trace-status** command to get updated information regarding packet trace.

Syntax: trace-status

Example:

```
trace-status
------------------------ Configuration -----------------------------
Trace Status:OFF  Wrap Mode:OFF  Decode Packets:OFF  HD Shadowing:OFF
RAM Trace Buffer Size:0  Maximum Trace Buffer File Size:10000000
Max Packet Bytes Trace:256  Default Packet Bytes Traced:100
Trace File Record Size:2048  Stop Trace Event: None
Maximum Hours to HD Shadow:  24
----------------------- Run-time Status ----------------------------
Packets in RAM Trace Buffer:0   Free Trace Buffer Memory:0
Trace Errors:0  First Packet:0  Last Packet:0
Trace Records Stored on HD:0  Trace Buffer File Size:0
HD-Shadowing Time Exceeded? NO
Has Stop Trace Event Occurred? NO
```

View

Use the **view** command to enter the View Captured Packet Trace Buffers Console.

Syntax: **view**

```
View Captured Packet Trace Buffers Console
ELS Packet Trace View>?
CURRENT
FIRST
JUMP
LAST
NEXT
PREV
SEARCH hexstring
EXIT
```

Example:

```
ELS PACKET TRACE View>current
```

For an explanation of the **view** commands, see “View” on page 9-19.

Chapter 10. Using and Configuring Bandwidth Reservation and Priority Queuing

This chapter describes the Bandwidth Reservation System and priority queuing features currently available for Frame Relay and PPP interfaces, and the commands for configuring them. It includes the following sections:

Bandwidth Reservation System

The Bandwidth Reservation System (BRS) allows you to decide which packets to drop when demand (traffic) exceeds supply (throughput) on a network connection. When bandwidth utilization reaches 100%, BRS determines which traffic to drop based on your configuration.

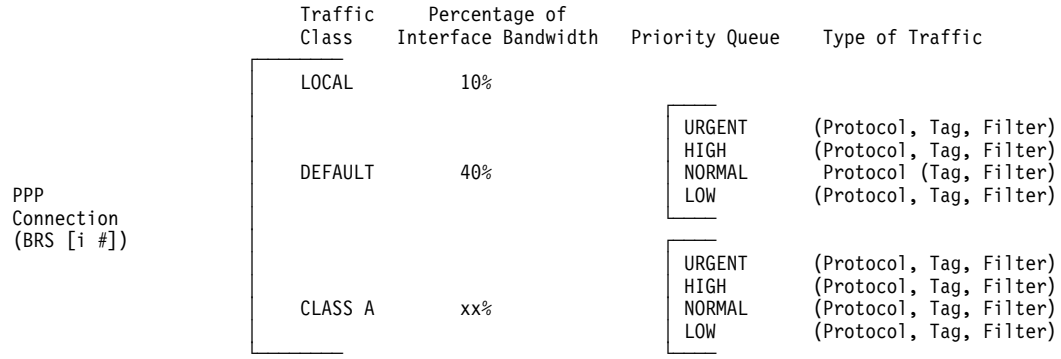
Bandwidth reservation "reserves" transmission bandwidth for specified classes of traffic. Each class has an allocated minimum percentage of the connection's bandwidth. See Figure 10-1 on page 10-2 and Figure 10-2 on page 10-2.

On PPP interfaces, you define traffic classes (t-classes) and each traffic class is allocated a percentage of the PPP interface's bandwidth. There are at least two traffic classes:

1. A LOCAL class which is allocated bandwidth for packets that are locally originated by the router (e.g. IP RIP packets)
2. A DEFAULT class to which all other traffic is initially assigned.

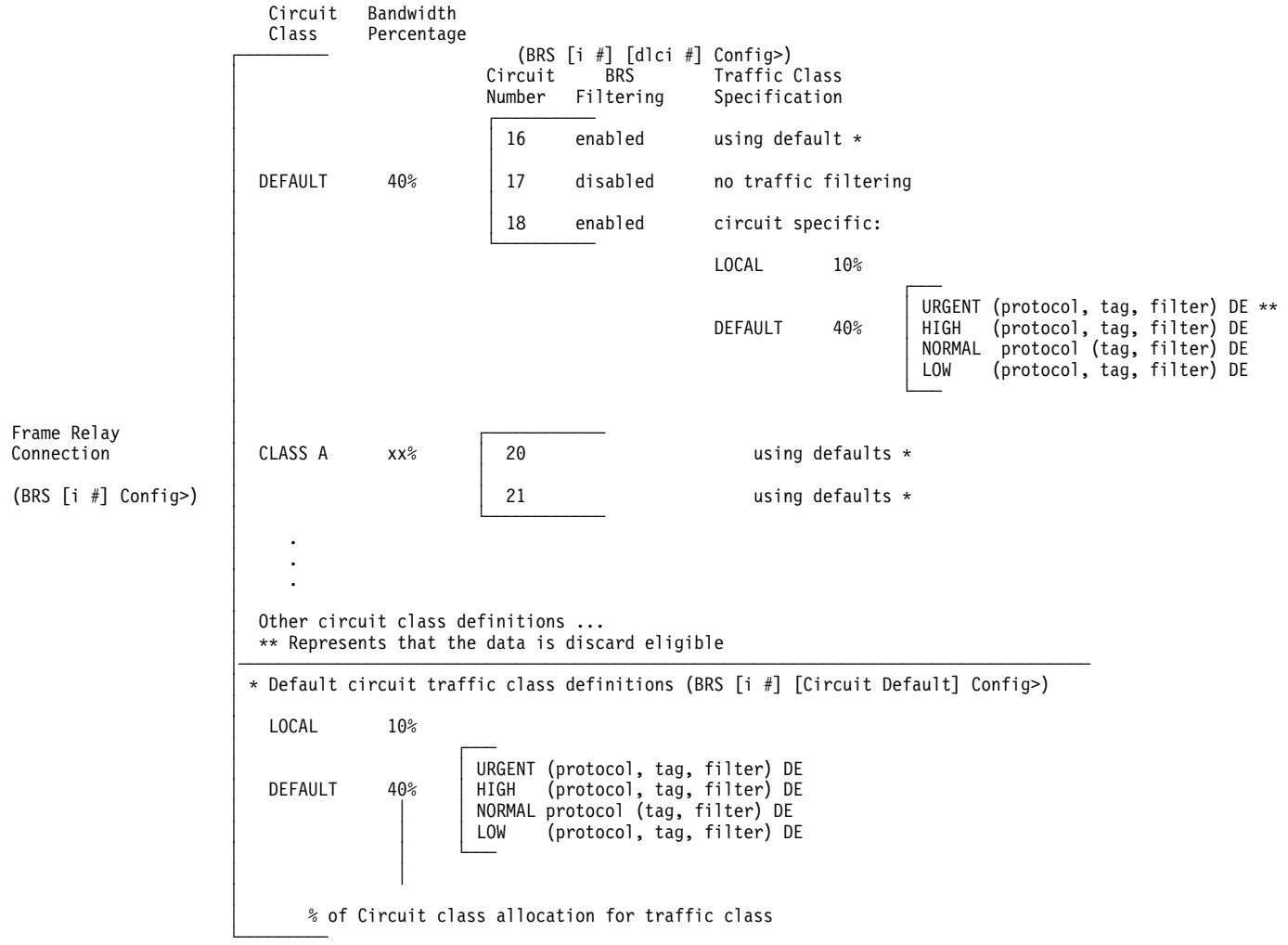
You can create additional traffic classes and assign protocols, filters and tags to the priority queues within a traffic class. See Figure 10-1 on page 10-2.

On Frame Relay interfaces, you define circuit classes (c-classes) and each circuit class is allocated a percentage of the Frame Relay interface's bandwidth. There is at least one circuit class: the DEFAULT circuit class to which all circuits are initially assigned. You can create additional circuit classes and assign circuits to these c-classes. On each Frame Relay circuit, you can define traffic classes (t-classes) and each traffic class is allocated a percentage of the Frame Relay circuit's bandwidth. The traffic class support for Frame Relay circuits is analogous to the traffic class support for PPP interfaces. See Figure 10-2 on page 10-2 for the Frame Relay Circuit Class and Traffic Class Relationships.



Note: All protocols are initially assigned to the NORMAL priority queue of the DEFAULT traffic class. You can assign a protocol, filter, or tag to any priority queue within a traffic class.

Figure 10-1. PPP BRS Traffic Class and Traffic Class Priority Queue Relationship



Note: All protocols are initially assigned to the NORMAL priority queue of the DEFAULT traffic class. You can assign a protocol, filter, or tag to any priority queue within a traffic class.

Figure 10-2. Frame Relay BRS Circuit Class and Traffic Class Relationship

These reserved percentages are a minimum *slice* of bandwidth for the network connection. If a network is operating to capacity, messages in any one class can be

transmitted only until they use the configured bandwidth allocated for the class. In this case, additional transmissions are held until other bandwidth transmissions have been satisfied. In the case of a light traffic path, a packet stream can use bandwidth exceeding its allowed minimum up to 100% if there is no other traffic.

Bandwidth reservation is really a *safeguard*. In general, a device should not attempt to use greater than 100% of its line speed. If it does, a faster line is probably needed. The “bursty” nature of traffic, however, can drive the requested transmission rate to exceed 100% for a short time. In these cases, bandwidth reservation is enabled and the higher priority traffic is ensured delivery (that is, is not discarded).

Bandwidth reservation runs over the following connection types:

- Frame Relay (serial line or dial circuit interface)
- PPP (serial line or dial circuit interface)

Bandwidth Reservation over Frame Relay

Bandwidth reservation allows you to reserve bandwidth at two levels:

- At the interface level, you can assign a percentage of the interface’s bandwidth to circuit classes (*c-classes*). Each circuit class contains one or more circuits.
- At the circuit level, you can define traffic classes and allocate a percentage of the circuit’s bandwidth.

Packets are filtered and queued into BRS t-classes based on the packet’s protocol type and any configured BRS filters. The packets are then queued into a BRS c-class based on the DLCI number.

The actual amount of bandwidth available for bandwidth reservation depends upon how you configure the interface and circuit:

- If you enable Frame Relay CIR monitoring, the bandwidth available to the circuit is allocated strictly according to its committed information rate (CIR), its committed burst size, and its excess burst size.
- If you disable CIR monitoring, up to 100 % of the bandwidth of the interface may be available to a circuit.

Orphaned circuits and circuits without BRS explicitly enabled use a default BRS queuing environment where the packets are queued on the default t-class and priority and the default c-class.

You can use several bandwidth reservation monitoring commands to display reservation counters for the circuit classes for a given interface:

- clear-circuit-class
- counters-circuit-class
- last-circuit-class

See Chapter 11, “Monitoring Bandwidth Reservation” on page 11-1 for more information on monitoring BRS.

The interface is the one shown at your prompt for the bandwidth monitoring commands. For example, `BRS [i 5]` is the prompt for interface 5.

If you do not want to use BRS circuit classes, leave all circuits in the default c-class and do not create any other circuit classes.

Queuing Support

With bandwidth reservation over Frame Relay, each circuit can queue frames while in the congested state, even for interfaces and circuits that are not enabled for bandwidth reservation.

Discard Eligibility

The Frame Relay network may discard transmitted data exceeding CIR on a PVC. The DE bit can be set by the router to indicate that some traffic should be considered discard eligible. If appropriate, the Frame Relay network will discard frames marked as discard eligible, which may allow frames that are not marked discard eligible to make it through the network. When assigning a protocol, filter, or tag to a traffic class, you can specify whether or not the protocol, filter, or tag traffic is discard eligible. See 10-15 for more information on how to configure traffic as discard eligible.

Default Circuit Definitions for Traffic Class Handling

Frame Relay interfaces can have many circuits defined. Rather than having to fully configure traffic class definitions for each circuit, BRS allows you to define a default set of traffic classes and protocol, filter, and tag assignments called default circuit definitions that can be used by any circuit on the interface. When BRS is initially enabled on a circuit, the circuit is initialized to use default circuit definitions. If a circuit cannot use the default circuit definitions for traffic class handling then you can create circuit specific definitions by using the **add-class**, **change-class**, **assign**, **deassign**,**tag**, and **untag** commands.

If a circuit is using circuit specific definitions and you want it to use the default circuit definitions instead, you can use the **use-circuit-defaults** command at the circuit's BRS prompt.

The default circuit definitions for traffic class handling are defined by using the **set-circuit-defaults** at the BRS Frame Relay interface prompt. This command gets you to a BRS circuit defaults prompt where you can add, change, and delete traffic classes, assign and deassign protocols, filters, and tags, and create BRS tags. Changes to the default circuit definitions for traffic classes result in dynamic updates to the traffic class handling for all circuits using the default circuit definitions.

Priority Queuing

Bandwidth reservation allocates percentages of total connection bandwidth for specified traffic *classes*, or *t-classes*, defined by the user. A BRS t-class is a group of packets identified by the same name; for example, a class called “ipx” to designate all IPX packets.

With priority queuing, each bandwidth t-class can be assigned one of the following priority level settings:

- Urgent
- High
- Normal (the default setting)

- Low

All packets assigned the Urgent priority are sent first within their class. These packets are followed by High, Normal, and then Low messages respectively. When all Urgent packets have been transmitted, High packets are transmitted until all are sent (or until new Urgent messages are queued). Only when there are no Urgent, High, or Normal packets remaining are the Low priority packets transmitted. If no priority setting is assigned, the setting defaults to Normal.

Also, you can set the number of packets that are waiting in the queue for each priority level in each bandwidth t-class. The BRS **queue-length** command sets the maximum number of output buffers that can be queued in each BRS priority queue, and the maximum number of output buffers that can be queued in each BRS priority queue for when router input buffers are scarce. You can set up priority queue lengths for both PPP and Frame Relay.

Attention: If you set the values for queue length too high, you may seriously degrade the performance of your router.

For BRS, you can set priority queue lengths for PPP and Frame Relay WAN connections. See “Queue-length” on page 10-24 for a description of the **queue-length** command.

The priority settings in one bandwidth t-class have no effect on other bandwidth classes. No one bandwidth class has priority over the others.

Priority Queuing Without Bandwidth Reservation

When priority queuing is configured without bandwidth reservation, the highest priority traffic is delivered first. In instances of heavy high-priority traffic, lower priority levels can be overlooked. By combining priority queuing with bandwidth reservation, however, packet transmission can be allocated to all types of traffic.

Configuring Traffic Classes

You create a traffic class using the **add-class** command and then assign types of traffic to the class using the **assign** command. Traffic is assigned to a traffic class based on its *protocol type* or based on a filter that further identifies a specific type of *protocol traffic* (for example, SNMP IP packets).

Supported protocol types are:

- IP
- ARP
- DNA
- VINES
- IPX
- OSI
- AP2
- ASRT
- SNA/APPN-ISR
- APPN-HPR

BRS Filters

Using bandwidth reservation, you can treat specific protocol traffic differently from other traffic that is using the same protocol type. For example, you can assign SNMP IP traffic to a different traffic class and priority than other IP traffic. In this example, SNMP is a BRS filter because it "filters" (i.e. uniquely identifies) specific protocol traffic. IP, ASRT (bridging) and APPN-HPR protocol traffic can be "filtered" by bandwidth reservation and the following filters are supported:

- IP tunneling
- SDLC tunneling over IP (SDLC Relay)
- Rlogin
- Telnet
- SNA/APPN-ISR
- APPN-HPR
- SNMP
- IP Multicast
- DLSw
- MAC Filter
- NetBIOS
- Network-HPR
- High-HPR
- Medium-HPR
- Low-HPR
- XTP
- TCP/UDP port numbers or sockets

BRS and Filtering

The following sections describe how to use BRS with various types of filtering.

MAC Address Filtering and Tags

MAC Address filtering is handled by a joint effort between bandwidth reservation and MAC filtering (MCF) using *tags*. For example, a user with bandwidth reservation is able to categorize bridge traffic by assigning a tag to it.

The tagging process is done by creating a filter item in the MAC filtering configuration console and then assigning a tag number to it. This tag number is used to set up a traffic class for all packets associated with this tag. Tag values must currently be in the range 1 through 64. See Chapter 12, “Using and Configuring MAC Filtering” on page 12-1 for additional information about MAC filtering.

Note: Tags can be applied ***only*** to bridged packets. On a PPP or Frame Relay connection, up to five tagged MAC filters can be assigned as bandwidth reservation filters and are designated as TAG1 through TAG5. TAG1 is searched for first, then TAG2, and so on up to TAG5. A single MAC filter tag can consist of any number of MAC Addresses set in MCF.

Once you have created a tagged filter in the MAC filtering configuration process, you can use the BRS tag configuration command to assign a BRS tag name (TAG1, TAG2, TAG3, TAG4, or TAG5) to the MAC filter tag number. Then use the BRS tag name on the BRS assign command to assign the corresponding MAC filter to a bandwidth traffic class and priority.

Tags also can refer to “groups,” as in the example of IP Tunnel. IP Tunnel endpoints can belong to any number of groups. Packets are assigned to a particular group through the tagging feature of MAC Address filtering. For additional information on MAC filtering, refer to Chapter 12, “Using and Configuring MAC Filtering” on page 12-1 and Chapter 13, “Monitoring MAC Filtering” on page 13-1.

To apply bandwidth reservation and queuing priority to tagged packets:

1. Use the MAC filtering configuration commands at the `filter config>` prompt to set up tags for packets passing through the bridge. Refer to Chapter 12, “Using and Configuring MAC Filtering” on page 12-1 for more information.
2. Use the bandwidth reservation **tag** command to reference a tag for bandwidth reservation.
3. With the bandwidth reservation **assign** command, assign the BRS tag to a t-class. The **assign** command also prompts you for a queuing priority within that BRS t-class.

TCP/UDP Port Number Filtering

You can assign TCP/IP packets from a range of TCP or UDP ports to a BRS t-class and priority based on the packet’s UDP or TCP port number and, optionally, upon a socket. You can specify up to 5 UDP/TCP port number filters, where the filters specify either an individual TCP or UDP port number, a range of TCP or UDP port numbers, or a socket identifier (combination of port number and IP address). You can then assign that filter to a BRS traffic class and priority within the class.

If UDP/TCP port filtering is enabled, BRS looks at each TCP or UDP packet and checks to see if the destination or source port number matches one of the port numbers you have specified for filtering. Also, if you define an IP address as part of the BRS UDP/TCP filter and the destination or source IP address matches the filter address you define, BRS assigns the packet to the traffic class and priority for that port number filter.

For example, you can configure a UDP port number filter for UDP port numbers in the range 25 to 29 and assign the filter to traffic class ’A’ with a priority of ’normal’. BRS queues any UDP packets with a source or destination port number from 25 to 29 on the normal priority queue for traffic class ’A’.

You can also configure a TCP port number filter for TCP port number 50 for IP address 5.5.5.25 and assign the filter to traffic class ’B’ with priority ’urgent’. BRS queues any TCP packets whose source or destination port number is 50 and whose destination or source IP address is 5.5.5.25 on the urgent priority queue for traffic class ’B’.

SNA and APPN Filtering

The SNA/APPN-ISR filter allows you to assign SNA and APPN-ISR traffic that is being bridged to a BRS traffic class. SNA and APPN-ISR traffic is identified as any bridge packets with a destination or source SAP of 0x04, 0x08, or 0x0C and whose LLC (802.2) control field indicates that it is not an unnumbered information (UI) frame.

Note: Frame Relay BAN packets are in this category.

The APPN-HPR filters allow you to assign HPR traffic that is being bridged to a BRS t-class. HPR traffic is identified as any bridge packet with a destination or source SAP of X'04', X'08', X'0C', or X'C8' and whose LLC (802.2) control field indicates it is an unnumbered information (UI) frame.

The Network-HPR, High-HPR, Medium-HPR, and Low-HPR filters allow HPR bridge traffic to further be filtered according to the HPR transmission priority. For example, if you want to assign HPR traffic that uses the network transmission priority to one t-class and priority and all other HPR bridged traffic to a different t-class or priority, you would assign the Network-HPR filter to the appropriate t-class and priority and use the APPN-HPR filter to assign the rest of the HPR traffic to a different t-class or priority.

Other filters may help you to assign traffic. For example, the DLSw filter allows you to assign SNA-DLSw traffic that is being sent over a TCP connection to a BRS t-class.

For SNA/APPN-ISR and APPN-HPR filters, if you want to check for SAPs other than the ones above, create a sliding window filter using MAC filtering and tag that filter. Then assign the tagged MAC filter to a BRS t-class.

Order of Filtering Precedence

It is possible for a packet to match more than one BRS filter type. For example, an IP tunneled bridge packet containing SNA data would match the IP tunneling filter and the SNA/APPN-ISR filter. The order in which the filters are evaluated to determine whether or not a packet matches a BRS filter type is as follows:

1. MAC filter tag match for bridging packets (IP/ASRT)
2. NetBIOS for bridging (IP/ASRT)
3. SNA/APPN-ISR for bridging (IP/ASRT)
4. HPR-Network (IP/ASRT/APPN-HPR)
5. HPR-High (IP/ASRT/APPN-HPR)
6. HPR-Medium (IP/ASRT/APPN-HPR)
7. HPR-Low (IP/ASRT/APPN-HPR)
8. APPN-HPR (IP/ASRT)
9. UDP/TCP port number filters (IP)
10. IP tunneling (IP)
11. SDLC relay (IP)
12. DLSw (IP)
13. Multicast (IP)
14. SNMP (IP)
15. Rlogin (IP)
16. Telnet (IP)
17. XTP (IP)

Note: The protocols for which a filter applies are shown in parentheses

Accessing the Bandwidth Reservation Configuration Prompt

To access bandwidth reservation configuration commands and configure bandwidth reservation on your router:

1. At the `OPCON (*)` prompt, enter **talk 6**.
2. At the `Config>` prompt, enter **feature brs**.

3. At the `BRS Config>` prompt, enter **interface #**.
4. At the `BRS [i 0] Config>` prompt, enter **enable**.

   This is the interface prompt level, and the interface number is zero in this instance. You need to repeat step 3 and step 4 for each interface you are configuring.

   If you are configuring BRS on a Frame Relay interface, continue with step 4a:

   If you are configuring BRS on any other interface, go directly to step 5.

   a. At the `BRS [i 0] Config>` prompt, enter **circuit** #, where # is the number of the circuit you want to configure.
   b. At the `BRS [i 0] [dlci 16] Config>` prompt, enter **enable**. This is the circuit prompt level and the circuit (DLCI) number is 16 in this instance.
   c. At the `BRS [i 0] [dlci 16] Config>` prompt, enter **exit** to return to the interface level prompt.
   d. Repeat steps 4a through 4c for each circuit for which you want to define BRS t-classes.
5. Restart your router.
6. Repeat steps 1 through 3 to configure bandwidth reservation for the particular interface that you have enabled.
7. If you are configuring BRS on a PPP interface, at the `BRS[i 0]Config>` prompt, configure traffic classes and assign protocols, filters, and tags to the traffic classes using the configuration commands listed in Table 10-3 on page 10-11. If you are configuring BRS on a FR interface, follow steps 8 through 10.
8. If you are configuring BRS on a FR interface, you can configure circuit classes and assign circuits to circuit classes using the commands listed in Table 10-2 on page 10-10
9. If you want to use default circuit definitions then enter the **set-circuit-defaults** command at the `BRS[i 0]Config>` prompt. This gets you to the `BRS[i 0][circuit defaults]` prompt where you can use the appropriate commands from Table 10-3 on page 10-11 to configure traffic classes and assign protocols, filters, and tags to the traffic classes. Once you are through defining default circuit definitions for traffic class handling, enter "exit" to return to the `BRS[i 0] Config>` prompt.
10. If you have FR circuits that cannot use default circuit definitions for traffic class handling, enter **circuit** *permanent-virtual-circuit circuit_number*. This will access the circuit prompt where you can use the commands listed in Table 10-3 on page 10-11 to create circuit-specific definitions for traffic class handling.

Note: You do not need to restart the router for t-class and c-class configuration changes to take effect.

The **talk 6 (t 6)** command lets you access the configuration process.

The **feature brs** command lets you access the BRS configuration process. You can enter this command by using either the feature name (brs) or number (1).

The **interface #** command selects the particular interface that you want to configure for bandwidth reservation. Before configuring any BRS classes, you must use the

enable command to enable BRS on the interface. In Step 4, the prompt indicates that the selected interface's number is zero.

The **circuit #** command selects the circuit on the FR interface on which you want to configure BRS traffic classes. Before configuring any BRS t-classes for the circuit, you must use the **enable** command to enable BRS on the circuit. In step 4b on page 10-9, the prompt indicates that circuit 16 on interface 0 has been selected.

You must enable bandwidth reservation for the selected interface and circuit and then restart your router before configuring circuit classes (Frame Relay only), and traffic classes.

To return to the `Config>` prompt at any time, enter the **exit** command at the different levels of BRS prompts until you are at the `Config>` prompt.

Bandwidth Reservation Configuration Commands

This section describes the Bandwidth Reservation configuration commands. The commands that can be used differ depending on the BRS configuration prompt that is displayed (`BRS Config>`, `BRS [i x] Config>`, or `BRS [i x] [dlci y] Config>`, or `BRS [i x] [circuit defaults] Config>`). See "? (Help)" on page 10-13 for a list of the commands supported at each BRS configuration prompt.

Table 10-1. Bandwidth Reservation Configuration Command Summary (Available from BRS Config> prompt)

Command	Function
? (Help)	Displays the Bandwidth Reservation configuration commands or lists the subcommand options for specific commands (if available).
Interface	Selects an interface on which to configure bandwidth reservation. **Note:** This command must be entered before using any other configuration commands. See Table 10-2 and Table 10-3 on page 10-11.
List	Lists the interfaces that can support bandwidth reservation and, for each interface, indicates if bandwidth reservation is enabled or disabled.
Exit	Exits the current bandwidth reservation prompt.

Table 10-2 (Page 1 of 2). BRS Interface Configuration Commands Available from BRS [i #] Config> prompt for Frame Relay Interfaces

Command	Function
? (Help)	Displays the Bandwidth Reservation configuration commands or lists the subcommand options for specific commands (if available).
Add-circuit-class	Sets the name of a bandwidth c-class and its percentage of bandwidth.
Assign-circuit	Assigns a specified circuit to the specified bandwidth c-class.
Change-circuit-class	Changes the amount of bandwidth configured for a bandwidth c-class.

Table 10-2 (Page 2 of 2). BRS Interface Configuration Commands Available from BRS [i #] Config> prompt for Frame Relay Interfaces

Command	Function
Circuit	Accesses the BRS circuit-level prompt (`BRS [i x][dlci y] Config>`) prompt where you can use the commands listed in Table 10-3 on page 10-11 to configure Bandwidth Reservation on the Frame Relay circuit.
Clear-block	Clears the configuration data associated with the current interface from SRAM. Circuit class configuration data and default circuit definitions for traffic class handling are cleared.
Deassign-circuit	Restores the specified circuit to the default c-class
Default-circuit-class	Assigns the name of a default bandwidth c-class and its percentage of the interface's bandwidth.
Del-circuit-class	Deletes the specified bandwidth c-class.
Disable	Disables bandwidth reservation on the interface .
Enable	Enables bandwidth reservation on the interface.
List	Displays the c-classes and assigned circuit definitions from SRAM.
Queue-length	Sets the maximum and minimum values for the number of packets in a priority queue.
Set-circuit-defaults	Accesses the `BRS [i x] [circuit defaults] Config>` command prompt so that you can use the appropriate commands from Table 10-3 on page 10-11 to create default circuit defintions for traffic class handling.
Show	Displays the currently defined c-classes and assigned circuits from SRAM.
Exit	Exits the current bandwidth reservation prompt.

The following table lists BRS circuit commands Available from `BRS [i x] Config>` for PPP interfaces, `BRS [i x] dlci [y] Config>` prompt for Frame Relay circuits, and from the `BRS [i x] [circuit defaults] Config>` prompt.

Table 10-3 (Page 1 of 2). BRS Traffic Class Handling Commands

Command	Function
? (Help)	Displays the Bandwidth Reservation configuration commands or lists the subcommand options for specific commands (if available).
Add-class	Allocates a designated amount of bandwidth to a user-defined traffic class.
Assign	Assigns a protocol or filter to a configured traffic class.
Change-class	Changes the amount of bandwidth configured for a bandwidth t-class.
Clear-block	Clears the traffic class and protocol, filter, and tag assignment configuration data from SRAM for the PPP interface or Frame Relay circuit. **Note:** This command cannot be used from the `BRS [i x] [circuit defaults] Config>` prompt.
Deassign	Restores the queuing of the specified packet or filter to the default t-class and priority.

Table 10-3 (Page 2 of 2). BRS Traffic Class Handling Commands

Command	Function
Default-class	Sets the default t-class and priority to a desired value and assigns all unassigned protocols to the new default t-class.
Del-class	Deletes a previously configured bandwidth t-class.
Disable	Disables bandwidth reservation on the PPP interface or Frame Relay circuit. **Note:** BRS cannot be enabled or disabled from the `BRS [i x] [circuit defaults] Config>` prompt.
Enable	Enables bandwidth reservation on the PPP interface or Frame Relay circuit. **Note:** BRS cannot be enabled or disabled from the `BRS [i x] [circuit defaults] Config>` prompt.
List	Lists the configured t-classes and protocol, filter and tag assignments stored in SRAM.
Queue-length	Sets the maximum and minimum values for the number of packets in a priority queue. **Note:** This command is not supported at the `BRS [i x] [circuit defaults] Config>` prompt.
Show	Displays the currently defined t-classes and protocol, filter, and tag assignments stored in RAM. **Note:** This command is not supported at the `BRS [i x] [circuit defaults] Config>` prompt.
Tag	Assigns a BRS tag name (TAG1-TAG5) to a MAC filter that has been tagged during the configuration of the MAC Filtering feature.
Untag	Removes the relationship between a BRS tag name (TAG1-TAG5) and a MAC filter that has been tagged during configuration of the MAC filtering feature.
Use-circuit-defaults	Allows the user to delete the circuit-specific definitions and use the circuit-defaults definitions for the traffic-class handling. This command is valid at the `BRS [i x] dlci [y] Config>` prompt for Frame Relay only. **Note:** The router must be restarted in order for the defaults to become operational.
Exit	Exits the current bandwidth reservation prompt.

Use the appropriate commands to configure bandwidth reservation for the Point-to-Point protocol (PPP) and Frame Relay. For Frame Relay, you need to configure the circuit and the network interface. For PPP, you only need to configure the network interface.

Notes:

1. When the **clear-block, disable**, **enable**, **list**, and **show** commands are issued from within the BRS interface menu, they affect or list the bandwidth reservation information configured for the selected interface. When these commands are issued from within the BRS circuit menu, only the Frame Relay bandwidth reservation information configured for the permanent virtual circuit (PVC) is affected or listed.
2. Before using the bandwidth reservation commands, keep the following in mind:

- You must use the **interface** command to select an interface before you use any other configuration commands. (BRS configuration enforces this.)
- The *Class-name* parameter is case-sensitive.
- To view the current *class-names*, use the **list** or **show** command.
- After you enable bandwidth reservation on an interface or circuit, you can add/delete/change circuit and traffic classes and assign circuits or protocols dynamically. The only commands that require a router restart before taking effect are the enable, disable, use-circuit-defaults, and clear-block commands.

3. You do not need to restart the router for t-class and c-class configuration changes to take effect.

? (Help)

Use the **?** (**help**) command to list the available commands from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

The following example show the output that is displayed at the `BRS Config>` prompt.

```
Config>f brs
Bandwidth Reservation User Configuration
?(HELP)
INTERFACE
LIST
EXIT
```

The following example shows the output that is displayed at the interface-level `BRS [i #] Config>` prompt when you are configuring for Frame Relay.

```
?(HELP)
ENABLE
DISABLE
SET-CIRCUIT-DEFAULTS
CIRCUIT
ADD-CIRCUIT-CLASS
DEL-CIRCUIT-CLASS
CHANGE-CIRCUIT-CLASS
DEFAULT-CIRCUIT-CLASS
ASSIGN-CIRCUIT
DEASSIGN-CIRCUIT
QUEUE-LENGTH
LIST
SHOW
CLEAR-BLOCK
EXIT
```

The following example shows the output that is displayed at the `BRS [i #] Config>` prompt for non-Frame-Relay interfaces, and at the circuit-level `BRS [i #] [dlci #] Config>` prompt for Frame-Relay interfaces.

```
?(HELP)
ENABLE
DISABLE
USE-CIRCUIT-DEFAULTS
ADD-CLASS
DEL-CLASS
CHANGE-CLASS
DEFAULT-CLASS
TAG
UNTAG
ASSIGN
DEASSIGN
QUEUE-LENGTH
LIST
SHOW
CLEAR-BLOCK
EXIT
```

Note: The **Use-circuit-defaults** command is not supported on PPP interfaces

The following example shows the output that is displayed at the set defaults-level `BRS [i #] [circuit defaults] Config>` prompt when you are configuring for Frame Relay

```
ADD-CLASS
DEL-CLASS
CHANGE-CLASS
DEFAULT-CLASS
TAG
UNTAG
ASSIGN
DEASSIGN
LIST
EXIT
```

.

Add-circuit-class

Note: Used only when configuring Frame Relay.

Use the **add-circuit-class** command at the interface level to allocate a designated amount of bandwidth to be used by the group of circuits assigned to the user-defined bandwidth c-class.

Syntax: add-circuit-class *class-name %*

Example: **`add-circuit-class alpha 10`**

Add-class

Use the **add-class** command to allocate a designated amount of bandwidth to a user-defined bandwidth t-class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax: add-class *class-name* or *class# %*

Example:

```
 BRS [i 1] [dlci 17] Config>
add-class
 This circuit is currently using circuit defaults...
 Are you sure you want to override the defaults ?(Yes or [No]):y
 Class name [DEFAULT]? CIRC17
 Percent bandwidth to reserve [10]?5
 BRS [i 1] [dlci 17] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, circuit number 17
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 65%
 total classes defined (counting one local and one default) 5


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
     protocol DNA with default priority is not discard eligible
     protocol VINES with default priority is not discard eligible
     protocol IPX with default priority is not discard eligible
     protocol OSI with default priority is not discard eligible
     protocol AP2 with default priority is not discard eligible
     protocol ASRT with default priority is not discard eligible


 class DEF1 has 5% bandwidth allocated
   protocol IP with priority NORMAL is not discard eligible.


 class DEF2 has 5% bandwidth allocated
  protocol ARP with priority NORMAL is not discard eligible.


 class CIRC171 has 5% bandwidth allocated
   no protocols or filters are assigned to this class.

 assigned tags:

 default class is DEFAULT with priority NORMAL
```

Assign

Use the **assign** command to assign specified tags, protocol packets, or filters to a given t-class and priority within that class. The four priority types include:

- Urgent
- High
- Normal (the default priority)
- Low.

The **assign** command also allows you to set the Discard-eligible (DE) bit for Frame Relay frames.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the

command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the BRS [i x][circuit defaults]Config> command prompt.

Syntax: assign *protocol-class or TAG or filter-class class-name or class#*

Example:

```
assign IPX test
priority <URGENT/HIGH/NORMAL/LOW>: [NORMAL]? low
protocol IPX maps to class test with priority LOW
Discard eligible <yes/no> [N]?
```

Assign-circuit

Note: Used only when configuring Frame Relay.

Use the **assign-circuit** command at the interface level to assign the specified circuit (DLCI) to the specified bandwidth c-class.

Note: You must use the **circuit** command to enable BRS on the DLCI and restart the router before you can use this command to assign the circuit to a circuit class.

Syntax: assign-circuit *# class name*

Example: **assign-circuit 16 pubs**

Change-circuit-class

Note: Used only when configuring Frame Relay.

Use the **change-circuit-class** command at the interface level to change the percentage of the bandwidth to be used by the group of circuits assigned to the specified c-class.

Syntax: change-circuit-class *class-name %*

Example: **change-circuit-class alpha 20**

Change-class

Use the **change-class** command to change the amount of bandwidth configured for a bandwidth t-class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the BRS [i x][circuit defaults]Config> command prompt.

Syntax: change-class *class-name or class# %*

Example: **change test 10**

Circuit

Note: Used only when configuring Frame Relay.

Use the **circuit** command to configure the DLCI of a Frame Relay permanent virtual circuit (PVC). This command can only be issued from the BRS interface configuration prompt (`BRS [i #] Config>`).

Syntax: circuit *permanent-virtual-circuit #*

Example: **`circuit 16`**

Before you can use the **add-class**, **assign**, **default-class**, **del-class**, **deassign**, or **change-class** commands, you must enable BRS on the circuit and restart the router. For example.

```
BRS [i 1] Config> circuit
Circuit to reserve bandwidth:  [16]

BRS [i 1 ] [dlci 16] Config> enable
```

After the **enable** command is issued for the Frame-Relay circuit and the router is restarted, the following configuration commands are available for the circuit:

add-class	deassign	enable	tag
assign	default-class	exit	untag
change-class	del-class	list	clear-block
disable	show	use-circuit-defaults	

Clear-block

Use the **clear-block** command to clear the current bandwidth reservation configuration data from SRAM.

- If you enter this command from the interface prompt for PPP, all BRS configuration data is cleared for the interface.
- If you enter this command from the interface prompt for Frame Relay, BRS is no longer enabled on the interface or on any circuits of the interface, and all circuit-class configuration data and default circuit definitions for traffic class handling are cleared. However, the traffic-class configuration data for each individual circuit is not cleared and is available if you re-enable BRS on the interface.
- To clear a circuit's traffic-class configuration data, you first enter the **circuit** command from the interface-level prompt and then the **clear-block** command from the circuit-level prompt. After you have cleared the traffic-class configuration data for each circuit, enter the **clear-block** command from the interface-level prompt to clear the circuit-class configuration data. The changes do not take effect until the router is restarted.

Syntax: clear-block

Example:

```
clear-block
You are about to clear BRS configuration information for this interface
Are you sure you want to do this (Yes or No): y
BRS [i 1] Config>
```

Default-circuit-class

Note: Used only when configuring Frame Relay.

Use the **default-circuit-class** command at the interface level to set the user-defined name of the default bandwidth c-class and the percentage of the bandwidth allocated to that class of circuits, including orphans, that are not assigned to a bandwidth c-class.

Syntax: default-circuit-class *class-name %*

Example: **`default-circuit-class group 10`**

Del-circuit-class

Note: Used only when configuring Frame Relay.

Use the **del-circuit-class** command at the interface level to delete the specified bandwidth c-class.

Syntax: del-circuit-class *class-name*

Example: **`del-circuit-class group`**

Default-class

Use the **default-class** command to set the default t-class and priority to a desired value. If no value has been previously assigned, system default values are used. Otherwise, the last previously assigned value is used.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax: default-class *class-name* or *class# priority*

Example: **`default-class test normal`**

Del-class

Use the **del-class** command to delete a previously configured bandwidth t-class from the specified interface or circuit.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax: del-class *class-name* or *class#*

Example: `del-class ipclass`

Deassign

Use the **deassign** command to restore the queuing of the specified protocol packet or filter to the default t-class and priority.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax: deassign *prot-class* or *filter-class*

Example: `deassign IP`

Deassign-circuit

Note: Used only when configuring Frame Relay.

Use the **deassign-circuit** command at the interface level to restore the queuing of the specified circuit to the default c-class.

Syntax: deassign-circuit #

Example: `deassign 16`

Disable

Use the **disable** command to disable bandwidth reservation on the interface (if entered from the interface prompt) or on the circuit (if entered from the circuit prompt). The changes do not take effect until the router is restarted.

To verify that bandwidth reservation is disabled, enter the **list** command.

Syntax: disable

Example: `disable`

Enable

Use the **enable** command to enable bandwidth reservation on the interface (if entered from the interface prompt) or the circuit (if entered from the circuit prompt). The changes do not take effect until the router is restarted.

Syntax: enable

Example: `enable`

Note:

- When configuring BRS on a PPP interface, issue the **enable** command at the interface prompt, and then restart the router before configuring any traffic classes and assigning protocols and filters to traffic classes.

- When BRS is initially enabled on a Frame Relay circuit, the circuit is initialized to use default circuit definitions for traffic class handling. Issue the **enable** command at the interface prompt and at the circuit prompt of each circuit for which you want to define traffic classes. Then restart the router before configuring circuit classes for the interface and traffic classes for each circuit. For example:

```
t 6
Gateway user configuration
Config>f brs
Bandwidth Reservation User Configuration
BRS Config>interface 1
BRS [i 1] Config>enable
Please  restart router for this command to take effect
BRS [i 1] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1
maximum queue length 10, minimum queue length 3
total bandwidth allocated 10%
total circuit classes defined (counting one default) 1


class DEFAULT has 10% bandwidth allocated
  no circuits are assigned to this class.

default class is DEFAULT

BRS [i 1] Config>circ 16
BRS [i 1] [dlci 16] Config>enable
Defaults are in effect for this circuit.
Please  restart router for this command to take effect.
BRS [i 1] [dlci 16] Config>ex
Please  restart router for this command to take effect.
BRS [i 1] [dlci 16] Config>
*rest
Are you sure you want to  restart the gateway? (Yes or [No]): y
```

Interface

Use the **interface** command to select the serial interface to which bandwidth reservation configuration commands will be applied. *Bandwidth reservation is supported on routers running PPP (Point-to-Point Protocol) and Frame Relay interfaces.*

Notes:

1. To enter bandwidth reservation commands for a new interface, this command must be entered ***before*** using any other bandwidth reservation configuration commands. If you have exited the bandwidth reservation prompt and wish to return to make bandwidth reservation changes to a previously configured interface, this command must again be entered first.
2. If WAN Restoral is used and BRS is configured on a primary interface, BRS should also be configured on the secondary interface. Typically when WAN Restoral is used, the secondary interface takes on the identity of the primary interface. This is not true for BRS; therefore, BRS needs to be configured on both the primary and secondary interfaces.

To enable Bandwidth Reservation on a particular interface, at the `BRS Config>` prompt, enter the number of the interface that supports the particular protocol or feature. You can then use the BRS **enable** configuration command as described in this chapter. After enabling the interface number, you must restartthe 2210 for the

command to take effect before you can make any other configuration changes to the interface.

Notes:

1. If you are configuring BRS on a Frame Relay interface, you can use the **circuit** command to select circuits and enable bandwidth reservation on those circuits before you restart the router.

Syntax: interface *interface#*

Example: `interface 2`

List

Use the **list** command to display currently defined bandwidth classes and their guaranteed percentage rates.

The **list** command and **show** command are similar. The **list** command displays the current SRAM definitions and the **show** command displays the current RAM definitions.

Syntax: list *interface#*

Example: `list`

Depending on the prompt at which you issue the **list** command, various outputs are displayed. You can issue the **list** command from the following prompts:

- `BRS [i 1] [dlci 16] Config>`
- `BRS [i 1] Config>`
- `BRS Config>`
- `BRS [i 1] [circuit defaults] Config>`

Note: When you use this command from a Frame Relay circuit prompt (`BRS [i x] [dlci y] Config>`) it indicates if the circuit is using default circuit definitions or circuit-specific definitions for traffic class handling. If the circuit is using default circuit definitions, the traffic class, protocol, filter, and tag assignments currently defined for default circuit definitions are displayed. However, if you want to alter the default circuit definitions, you need to get to the `BRS[i x] [circuit defaults] Config>` prompt to make changes.

At the BRS interface level prompt (`BRS [i 0]`) for PPP interfaces and at the BRS circuit level prompt (`BRS [i 0] [dlci 16] Config>`) for Frame Relay interfaces, the **list** command lists the traffic classes, their configured bandwidth percentages, and the assigned protocols and filters.

At the BRS interface level prompt for Frame Relay, the **list** command lists the circuit classes, their configured bandwidth percentages, and the assigned circuits.

Example 1

```
BRS Config>list
 Bandwidth Reservation is available for 2 interfaces.

 Interface   Type            State
 ---------   ----            -----
         1   FR              Enabled
         2   PPP             Enabled

 BRS Config>interface 1
 BRS [i 1] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 10%
 total circuit classes defined (counting one default) 1


 class DEFAULT has 10% bandwidth allocated
   the following circuits are assigned:
       17
       16  using defaults.
       18  using defaults.

 default class is DEFAULT

 BRS [i 2] Config>exit
 BRS Config>interface 2
 BRS [i 2] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 2
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 50%
 total classes defined (counting one local and one default) 2


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
     protocol IP with default priority
     protocol ARP with default priority
     protocol DNA with default priority
     protocol VINES with default priority
     protocol IPX with default priority
     protocol OSI with default priority
     protocol AP2 with default priority
     protocol ASRT with default priority

 assigned tags:

 default class is DEFAULT with priority NORMAL

 BRS [i 2] Config>
```

Example 2

```
 BRS [i 1] [dlci 17] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
maximum queue length 10, minimum queue length 3
total bandwidth allocated 60%
total classes defined (counting one local and one default) 3


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
   protocol ASRT with priority NORMAL is not discard eligible
   filter NETBIOS with priority NORMAL is not discard eligible


class CLASS1 has 10% bandwidth allocated
  the following protocols and filters are assigned:
   protocol IP with priority NORMAL is not discard eligible
   protocol ARP with priority NORMAL is not discard eligible
   protocol DNA with priority NORMAL is not discard eligible
   protocol VINES with priority NORMAL is not discard eligible
   protocol IPX with priority NORMAL is discard eligible
   protocol OSI with priority NORMAL is not discard eligible
   protocol AP2 with priority NORMAL is not discard eligible
```

Example 3

```
 BRS [i 1] [circuit defaults] Config>
list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, default circuit
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 70%
 total classes defined (counting one local and one default) 4


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.



 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
     protocol DNA with default priority is not discard eligible
     protocol VINES with default priority is not discard eligible
     protocol IPX with default priority is not discard eligible
     protocol OSI with default priority is not discard eligible
     protocol AP2 with default priority is not discard eligible
     protocol ASRT with default priority is not discard eligible


 class DEF1 has 10% bandwidth allocated
   protocol IP with priority NORMAL is not discard eligible.


 class DEF2 has 10% bandwidth allocated
  protocol ARP with priority NORMAL is not discard eligible.

 assigned tags:

 default class is DEFAULT with priority NORMAL

  BRS [i 1] [circuit defaults] Config>
```

Queue-length

Use the **queue-length** command to set the number of packets that can be queued in each BRS priority queue. Each BRS class has a priority value assigned to its protocols, filters, and tags, and each priority queue can store the number of packets that you specify with this command.

This command sets the maximum number of buffers that can be queued in each BRS priority queue as well as the maximum number that can be queued in each BRS priority queue when there is a shortage of router input buffers.

If you issue **queue-length** for a PPP interface, the command sets the queue-length values for each priority queue of each BRS t-class that is defined for the interface.

If you issue **queue-length** for a Frame Relay interface (at the prompt: `BRS [i 0] Config>`), the command sets the default queue-length values for each priority queue of each BRS t-class that is defined for each permanent virtual circuit of the interface.

If you issue **queue-length** for a Frame-Relay PVC (at a prompt like this: `BRS [i 0] [dlci 16] Config>`) the command sets the queue length values for each priority queue of each BRS t-class that is defined for the PVC. These values override the default queue length values set for the Frame Relay interface.

Attention: Do not use this command unless it is essential to do so. The default values for queue length are the recommended values for most users. If you set the values for queue length too high, you may seriously degrade the performance of your router.

Syntax: queue-length *maximum-length minimum-length*

Example: **`queue-length`**

```
BRS priority queue maximum length [10]?
BRS priority queue minimum length [3]
```

Set-circuit-defaults

Use the **set-circuit-defaults** command to access the commands used to define default circuit definitions for traffic class handling. These default circuit definitions can then be used by any Frame Relay circuits on the interface that can use the same traffic classes and protocol, filter, and tag assignments.

Syntax: set-circuit-defaults

Example:

```
BRS [i 1] Config>set-circuit-defaults
BRS [i 1] [circuit defaults] Config>
```

Show

Use the **show** command to display currently defined bandwidth classes stored in RAM.

Syntax: show *interface#*

Example: **`show`**

Depending on the prompt at which you issue the **show** command, various outputs are displayed. You can issue the **show** command from the following prompts:

- BRS [i *x*] Config> - interface level prompt for interface number *x*.
- BRS [i *x*] [dlci *y*] Config> - circuit level prompt for circuit *y* on Frame Relay interface number *x*. The following example shows the output of the show command from the circuit level prompt.

```
BRS [i 1] [dlci 17] Config>show

Protocol/Filter      Class              Priority    Discard Eligible
---------------      -----              --------    ----------------
IP                   CLASS1               NORMAL             NO
ARP                  CLASS1               NORMAL             NO
DNA                  CLASS1               NORMAL             NO
VINES                CLASS1               NORMAL             NO
IPX                  CLASS1               NORMAL             YES
OSI                  CLASS1               NORMAL             NO
AP2                  CLASS1               NORMAL             NO
ASRT                 DEFAULT              NORMAL             NO
NETBIOS              DEFAULT              NORMAL             NO
```

At the interface prompt for PPP and the circuit prompt for Frame Relay, traffic class information is displayed. At the interface prompt for Frame Relay, circuit class information is displayed.

Notes:

1. When you use this command from a Frame Relay circuit prompt (BRS [i x] [dlci y] Config>) it indicates if the circuit is using default circuit definitions or circuit-specific definitions for traffic class handling. If the circuit is using default circuit definitions, the traffic class, protocol, filter, and tag assignments currently defined for default circuit definitions are displayed. However, if you want to alter the default circuit definitions, you need to get to the BRS [i x] [circuit defaults] Config> prompt to make changes.
2. This command cannot be used from the BRS [i x] [circuit defaults] Config> prompt.

Tag

Use the **tag** command to assign a MAC filter item that has been tagged during the configuration of the MAC filtering feature to the next available BRS tag name. The BRS tag names are TAG1, TAG2, TAG3, TAG4, and TAG5. You use the BRS tag name on the assign command to assign the tag to a BRS traffic class.

Use the **list** command to list which MAC filter tags have been assigned to a BRS tag name and which BRS tag names have been assigned to a bandwidth traffic class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the BRS [i x][circuit defaults]Config> command prompt.

Syntax: tag *mac_filter_tag#*

Example: **`tag 3`**

Untag

Use the **untag** command to remove the MAC filter tag number and BRS tag name relationship. A tag can be removed only if its corresponding BRS tag name is not assigned to a bandwidth traffic class.

Use the **list** command to show which MAC filter tags are assigned to a BRS tag name and which BRS tag names are assigned to a traffic class.

Note: If this command is used for a Frame Relay circuit that is currently using default circuit definitions for traffic class handling, you will be asked whether or not you want to override the default circuit definitions. If you answer "Yes," the circuit will be changed to use circuit-specific definitions for traffic class handling and the command will be allowed. If you answer "No," the command is aborted and default circuit definitions will continue to be used for the circuit. If you want to change the default circuit definitions, you should go to the `BRS [i x][circuit defaults]Config>` command prompt.

Syntax: untag *mac_filter_ag#*

Example: **`untag 3`**

Use-circuit-defaults

Use the **use-circuit-defaults** command at the circuit level to delete the circuit-specific definitions and use the circuit default definitions for traffic-class handling. You will be prompted to confirm that you want to use the circuit defaults.

Notes:

1. This command is used only when configuring Frame Relay
2. The router must be restarted for the defaults to become operational.

Syntax: use-circuit-defaults

Example:

```
BRS [i 1] [dlci 17] Config>use-circuit-defaults
This circuit is currently NOT using circuit defaults...
 Are you sure you want to delete current definitions and use defaults ? (Yes or
 [No]): y
 Defaults are in effect for this circuit.
 Please  restart router for this command to take effect.
 BRS [i 1] [dlci 17] Config>
 *rest
 Are you sure you want to  restart the gateway? (Yes or [No]): y
```

Exit

Use the **exit** command to return to the previous level prompt. For example, entering the **exit** command at the circuit level prompt (for example, `BRS [i 0] [dlci 16] Config>`) takes you back to the interface level prompt (`BRS [i 0] Config>`). Entering the **exit** command at the interface level prompt returns you to the general BRS prompt (`BRS Config>`), and entering the **exit** command at the general BRS prompt returns you to the `Config>` prompt.

Syntax: exit

Example: `exit`

Sample Configurations

Using Default Circuit Definitions for Traffic Class Handling of Frame Relay Circuits

Notes:

1. Configure feature BRS.
2. Enable BRS on interface 1.
3. Enable BRS on circuits 16, 17, 18. Default circuit definitions for traffic class handling are used for these circuits.
4. Access the set-circuit-defaults menu to define default circuit definitions for traffic class handling.
5. Add traffic classes and assign protocols and filters to the traffic classes.
6. List and show the BRS definitions for circuit 16. Since circuit 16 is using default circuit definitions, the traffic classes and protocol and filter assignments defined by the default circuit definitions are displayed.
7. Change circuit 17 from using default circuit definitions to use circuit-specific definitions for traffic class handling by creating a unique class, CIRC171. This class can have protocols, filters, or tags assigned to it.
8. Change the default circuit definitions such that the DEF1 and DEF2 traffic classes each reserve 10% of the bandwidth and then show that these changes are picked up by circuit 16 but not by circuit 17, since circuit 17 is now using circuit-specific definitions.
9. Alter circuit 17 to use default circuit definitions for traffic class handling instead of circuit-specific definitions.

```
t 6
 Gateway user configuration
 Config>feature brs      1
 Bandwidth Reservation User Configuration
 BRS Config>interface 1     2
 BRS [i 1]Config>enable
 Please restart router for this command to take effect.
 BRS [i 1] Config>circuit 16    3
 BRS [i 1][dlci 16] Config>enable
 Defaults are in effect for this circuit.
 Please restart router for this command to take effect.
 BRS [i 1][dlci 16] Config>exit
 BRS [i 1]Config>circuit 17
 BRS [i 1][dlci 17] Config>enable
 Defaults are in effect for this circuit.
 Please restart router for this command to take effect.
 BRS [i 1][dlci 17] Config>exit
 BRS [i 1]Config>circuit 18
 BRS [i 1][dlci 18] Config>enable
 Defaults are in effect for this circuit.
 Please restart router for this command to take effect.
 BRS [i 1][dlci 18] Config>
 *restart
 Are you sure you want to restart the gateway? (Yes or [No]): yes
```

```
 *t 6
 Gateway user configuration
 Config>feature brs
 Bandwidth Reservation User Configuration
 BRS Config>interface 1
 BRS[i 1] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 10%
 total circuit classes defined (counting one default) 1


 class DEFAULT has 10% bandwidth allocated
   the following circuits are assigned:
       16  using defaults.
       17  using defaults.
       18  using defaults.

 default class is DEFAULT
```

```
BRS [i 1] Config>?
 ENABLE
 DISABLE
 SET-CIRCUIT-DEFAULTS
 CIRCUIT
 ADD-CIRCUIT-CLASS
 DEL-CIRCUIT-CLASS
 CHANGE-CIRCUIT-CLASS
 DEFAULT-CIRCUIT-CLASS
 ASSIGN-CIRCUIT
 DEASSIGN-CIRCUIT
 QUEUE-LENGTH
 LIST
 SHOW
 CLEAR-BLOCK
 EXIT
 BRS [i 1] Config>set-circuit-defaults     4
 BRS [i 1] [circuit defaults] Config>?
 ADD-CLASS
 DEL-CLASS
 CHANGE-CLASS
 DEFAULT-CLASS
 TAG
 UNTAG
 ASSIGN
 DEASSIGN
 LIST
 EXIT
 BRS [i 1] [circuit defaults] Config>add    5
 Class name [DEFAULT]?DEF1
 Percent bandwidth to reserve [10]? 5
 BRS [i 1] [circuit defaults] Config>add
 Class name [DEFAULT]?DEF2
 Percent bandwidth to reserve [10]?5
 BRS [i 1] [circuit defaults] Config>assign ip
 Class name [DEFAULT]?DEF1
 Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]?
 Frame Relay Discard Eligible <NO/YES> [NO]?
 BRS [i 1] [circuit defaults] Config>assign asrt
 Class name [DEFAULT]? DEF2
 Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]?
 Frame Relay Discard Eligible <NO/YES> [NO]?
 BRS[i 1] [circuit defaults] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, default circuit
 total bandwidth allocated 60%
 total classes defined (counting one local and one default) 4


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.
```

```
class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL

BRS  [i 1] [circuit defaults] Config>exit
BRS  [i 1] Config>circuit 16   6
BRS  [i 1][dlci 161] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, circuit number 16 using defaults.
maximum queue length 10, minimum queue length 3
total bandwidth allocated 60%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 5% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL
```

```
BRS  [i 1] [dlci 16] Config>show

 BANDWIDTH RESERVATION currently in RAM
 interface number 1, circuit number 16 using defaults.
 maximum queue length 10, minimum queue length 3
 4 current defined classes:
   class LOCAL has 10% bandwidth allocated
   class DEFAULT has 40% bandwidth allocated
   class DEF1 has 5% bandwidth allocated
   class DEF2 has 5% bandwidth allocated

 protocol and filter assignments:

 Protocol/Filter      Class              Priority    Discard Eligible
 ---------------      -----              --------    ----------------
 IP                   DEF1                 NORMAL             NO
 ARP                  DEFAULT              NORMAL             NO
 DNA                  DEFAULT              NORMAL             NO
 VINES                DEFAULT              NORMAL             NO
 IPX                  DEFAULT              NORMAL             NO
 OSI                  DEFAULT              NORMAL             NO
 AP2                  DEFAULT              NORMAL             NO
 ASRT                 DEF2                 NORMAL             NO

 BRS [i 1] [dlci 16] Config>exit
```

```
BRS [i 1] Config>circuit 17
 BRS [i 1] [dlci 17] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, circuit number 17 using defaults.
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 60%
 total classes defined (counting one local and one default) 4


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
     protocol ARP with default priority is not discard eligible
     protocol DNA with default priority is not discard eligible
     protocol VINES with default priority is not discard eligible
     protocol IPX with default priority is not discard eligible
     protocol OSI with default priority is not discard eligible
     protocol AP2 with default priority is not discard eligible

 class DEF1 has 5% bandwidth allocated
   the following protocols and filters are assigned:
     protocol IP with priority NORMAL is not discard eligible


 class DEF2 has 5% bandwidth allocated
   the following protocols and filters are assigned:
     protocol ASRT with priority NORMAL is not discard eligible

 assigned tags:

 default class is DEFAULT with priority NORMAL

 BRS [i 1] [dlci 17] Config>add-class    7
 This circuit is currently using circuit defaults...
 Are you sure you want to override the defaults ?(Yes or [No]): yes
 Class name [DEFAULT]? CIRC171
 Percent bandwidth to reserve [10]? 5
 BRS[i 1] [dlci 17] Config>assign vines
 Class name [DEFAULT]? CIRC171
 Priority <URGENT/HIGH/NORMAL/LOW> [NORMAL]?
 Frame Relay Discard Eligible <NO/YES>[NO]?
```

```
BRS [i 1] [dlci 17] Config>list

 BANDWIDTH RESERVATION listing from SRAM
 bandwidth reservation is enabled
 interface number 1, circuit number 17
 maximum queue length 10, minimum queue length 3
 total bandwidth allocated 65%
 total classes defined (counting one local and one default) 5


 class LOCAL has 10% bandwidth allocated
   protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
   the following protocols and filters are assigned:
     protocol ARP with default priority is not discard eligible
     protocol DNA with default priority is not discard eligible
     protocol IPX with default priority is not discard eligible
     protocol OSI with default priority is not discard eligible
     protocol AP2 with default priority is not discard eligible


 class DEF1 has 5% bandwidth allocated
   the following protocols and filters assigned:
     protocol IP with priority NORMAL is not discard eligible


 class DEF2 has 5% bandwidth allocated
   the following protocols and filters are assigned:
     protocol ASRT with priority NORMAL is not discard eligible


 class CIRC171 has 5% bandwidth allocated
   the following protocols and filters are assigned:
     protocol VINES with priority NORMAL is not discard eligible

 assigned tags:

 default class is DEFAULT with priority NORMAL

 BRS [i 1] [dlci 17] Config>show

 BANDWIDTH RESERVATION currently in RAM
 interface number 1, circuit number 17
 maximum queue length 10, minimum queue length 3
 5 current defined classes:
   class LOCAL has 10% bandwidth allocated
   class DEFAULT has 40% bandwidth allocated
   class DEF1 has 5% bandwidth allocated
   class DEF2 has 5% bandwidth allocated
   class CIRC171 has 5% bandwidth allocated
```

```
protocol and filter assignments:

Protocol/Filter      Class              Priority    Discard Eligible
---------------      -----              --------    ----------------
IP                   DEF1                 NORMAL             NO
ARP                  DEFAULT              NORMAL             NO
DNA                  DEFAULT              NORMAL             NO
VINES                CIRC171              NORMAL             NO
IPX                  DEFAULT              NORMAL             NO
OSI                  DEFAULT              NORMAL             NO
AP2                  DEFAULT              NORMAL             NO
ASRT                 DEF2                 NORMAL             NO

BRS [i 1] [dlci 17] Config>exit
BRS [i 1] Config>set-circuit-defaults
BRS [i 1] [circuit defaults] Config>change DEF1    8
Percent bandwidth to reserve [ 5]? 10
BRS [i 1] [circuit defaults] Config>change DEF2
Percent bandwidth to reserve [5]? 10
BRS [i 1] [circuit defaults] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, default circuit
total bandwidth allocated 70%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL

BRS [i 1] [circuit defaults] Config>exit
```

```
BRS [i 1] Config>circuit 16
  BRS [i 1] [dlci 16] Config>list

  BANDWIDTH RESERVATION listing from SRAM
  bandwidth reservation is enabled
  interface number 1, circuit number 16 using defaults.
  maximum queue length 10, minimum queue length 3
  total bandwidth allocated 70%
  total classes defined (counting one local and one default) 4


  class LOCAL has 10% bandwidth allocated
    protocols and filters cannot be assigned to this class.


  class DEFAULT has 40% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ARP with default priority is not discard eligible
      protocol DNA with default priority is not discard eligible
      protocol VINES with default priority is not discard eligible
      protocol IPX with default priority is not discard eligible
      protocol OSI with default priority is not discard eligible
      protocol AP2 with default priority is not discard eligible

  class DEF1 has 10% bandwidth allocated
    the following protocols and filters are assigned:
      protocol IP with priority NORMAL is not discard eligible


  class DEF2 has 10% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ASRT with priority NORMAL is not discard eligible

  assigned tags:

  default class is DEFAULT with priority NORMAL

  BRS [i 1] [dlci 16] Config>exit
```

```
BRS [i 1] Config>circuit 17
  BRS [i 1[ [dlci 17] Config>list

  BANDWIDTH RESERVATION listing from SRAM
  bandwidth reservation is enabled
  interface number 1, circuit number 17
  maximum queue length 10, minimum queue length 3
  total bandwidth allocated 65%
  total classes defined (counting one local and one default) 5


  class LOCAL has 10% bandwidth allocated
    protocols and filters cannot be assigned to this class.


  class DEFAULT has 40% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ARP with default priority is not discard eligible
      protocol DNA with default priority is not discard eligible
      protocol IPX with default priority is not discard eligible
      protocol OSI with default priority is not discard eligible
      protocol AP2 with default priority is not discard eligible


  class DEF1 has 5% bandwidth allocated
    the following protocols and filters are assigned:
      protocol IP with priority NORMAL is not discard eligible


  class DEF2 has 5% bandwidth allocated
    the following protocols and filters are assigned:
      protocol ASRT with priority NORMAL is not discard eligible


  class CIRC171 has 5% bandwidth allocated
    the following protocols and filters are assigned:
      protocol VINES with priority NORMAL is not discard eligible

  assigned tags:

  default class is DEFAULT with priority NORMAL

  BRS [i 1] [dlci 17] Config>use-circuit-defaults     9
  This circuit is currently NOT using circuit defaults...
  Are you sure you want to delete current definitions and use defaults ? (Yes or
  [No]): yes
  Defaults are in effect for this circuit.
  Please restart router for this command to take effect.
  BRS [i 1] [dlci 17] Config>
  *restart
  Are you sure you want to restart the gateway? (Yes or [No] ):yes
```

```
*t 6
Gateway user configuration
Config>feature brs
Bandwidth Reservation User Configuration
BRS Config>interface 1
BRS [i 1] Config>circuit 17
BRS [i 1] [dlci 17] Config>list

BANDWIDTH RESERVATION listing from SRAM
bandwidth reservation is enabled
interface number 1, circuit number 17 using defaults.
maximum queue length 10, minimum queue length 3
total bandwidth allocated 70%
total classes defined (counting one local and one default) 4


class LOCAL has 10% bandwidth allocated
  protocols and filters cannot be assigned to this class.


 class DEFAULT has 40% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ARP with default priority is not discard eligible
    protocol DNA with default priority is not discard eligible
    protocol VINES with default priority is not discard eligible
    protocol IPX with default priority is not discard eligible
    protocol OSI with default priority is not discard eligible
    protocol AP2 with default priority is not discard eligible

class DEF1 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    protocol IP with priority NORMAL is not discard eligible


class DEF2 has 10% bandwidth allocated
  the following protocols and filters are assigned:
    protocol ASRT with priority NORMAL is not discard eligible

assigned tags:

default class is DEFAULT with priority NORMAL


BRS [i 1] [dlci 17] Config>show

BANDWIDTH RESERVATION currently in RAM
interface number 1, circuit number 17 using defaults.
maximum queue length 10, minimum queue length 3
4 current defined classes:
  class LOCAL has 10% bandwidth allocated
  class DEFAULT has 40% bandwidth allocated
  class DEF1 has 10% bandwidth allocated
  class DEF2 has 10% bandwidth allocated

protocol and filter assignments:

Protocol/Filter      Class              Priority    Discard Eligible
---------------      -----              --------    ----------------
IP                   DEF1                 NORMAL            NO
ARP                  DEFAULT              NORMAL            NO
DNA                  DEFAULT              NORMAL            NO
VINES                DEFAULT              NORMAL            NO
IPX                  DEFAULT              NORMAL            NO
OSI                  DEFAULT              NORMAL            NO
AP2                  DEFAULT              NORMAL            NO
ASRT                 DEF2                 NORMAL            NO

BRS [i 1] [dlci 17] Config>exit
```

Chapter 11. Monitoring Bandwidth Reservation

This chapter describes how to access the Bandwidth Reservation System (BRS) console prompt and the available console commands.

This chapter includes the following sections:

- "Accessing the Bandwidth Reservation Console Prompt"
- "Bandwidth Reservation Console Commands"

Accessing the Bandwidth Reservation Console Prompt

To access bandwidth reservation console commands and to monitor bandwidth reservation on your router, do the following:

1. At the OPCON prompt (*), type **talk 5**.
2. At the GWCON prompt (+), type **feature brs.**
3. At the BRS> prompt, type **interface #**, where # is the number of the interface that you want to monitor. This takes you to the BRS interface-level prompt, BRS [i x]>, where x is the number of the interface number.
4. For Frame Relay only, type **circuit #** at the interface prompt to specify the circuit on this interface that you want to monitor.

   This takes you to the circuit-level prompt BRS [i x] [dlci y]>, where x is the interface number and y is the circuit number.
5. At the prompt, type the appropriate monitoring command. (Refer to "Bandwidth Reservation Console Commands.")

   The **talk 5 (t 5)** command lets you access the monitoring process.

   The **feature brs** command lets you access the BRS monitoring process. You can enter this command by using either the feature name (brs) or number (1).

   The **interface #** command selects the particular interface that you want to monitor for bandwidth reservation.

   The **circuit #** command selects the DLCI of a Frame Relay permanent virtual circuit (PVC).

   To return to the GWCON prompt at any time, type the **exit** command at the BRS> prompt.

   Once you access the bandwidth reservation console prompt (BRS>), you can enter any of the specific console commands described in Table 11-1 on page 11-2.

Bandwidth Reservation Console Commands

This section summarizes and explains the Bandwidth Reservation console commands. 11-1 shows the Bandwidth Reservation console commands. The commands that can be used differ depending on the BRS console prompt (BRS>, BRS [i x]>, or BRS [i x] [dlci y]>). See "? (Help)" on page 11-2 for a list of the console commands supported at each BRS console prompt.

Table 11-1. Bandwidth Reservation Console Command Summary

Command	Used Only With FR	Function
? (Help)		Displays the Bandwidth Reservation configuration commands or lists the subcommand options for specific commands (if available).
Circuit	yes	Selects the DLCI of a Frame Relay permanent virtual circuit (PVC). To monitor Frame Relay bandwidth reservation traffic, you must be at the circuit prompt level.
Clear		Clears the current t-class counters and stores them as **last** t-class counters. Counters are listed by class.
Clear-circuit-class	yes	Clears the current c-class counters and stores them as **last** c-class counters. Counters are listed by class.
Counters		Displays the current t-class counters.
Counters-circuit-class	yes	Displays the current c-class counters.
Interface		Selects the interface to monitor. **Note:** This command must be entered before using any other bandwidth reservation console commands.
Last		Displays the last saved t-class counters.
Last-circuit-class	yes	Displays the last saved c-class counters.
Exit		Exits the bandwidth reservation console process.

? (Help)

Use the **?** (**help**) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: ?

At the `BRS>` prompt:

```
INTERFACE
EXIT
```

For Frame Relay, at the `BRS [i #] [dlci #]>` prompt or for PPP, at the `BRS [i #]` prompt, enter:

```
CLEAR
COUNTERS
LAST
EXIT
```

For Frame Relay, at the `BRS [i #]>` prompt, enter: “Bandwidth Reservation Console Commands” on page 11-1.

```
CIRCUIT
CLEAR-CIRCUIT-CLASS
COUNTERS-CIRCUIT-CLASS
LAST-CIRCUIT-CLASS
EXIT
```

Circuit

Note: Used only when monitoring Frame Relay.

Use the **circuit** command to select the DLCI of a Frame Relay PVC for monitoring. This command can be issued only from the BRS interface monitoring prompt (`BRS [i #]>`).

Syntax: circuit *permanent virtual circuit #*

Example: **`circuit 16`**

After the Frame Relay circuit has been selected, the following commands can be used at the circuit prompt:

```
CLEAR
COUNTERS
LAST
EXIT
```

Clear

Use the **clear** command to save the current bandwidth reservation t-class counters so that they can be retrieved using the **last** command and clear the values. The counters are kept on a bandwidth traffic class basis.

Syntax: clear

Example: **`clear`**

Clear-Circuit-Class

Note: Used only when monitoring Frame Relay.

Use the **clear-circuit-class** command to save the current bandwidth reservation c-class counters so that they can be retrieved using the **last-circuit-class** command and clear the values. The counters are kept on a circuit class basis.

Syntax: clear

Example: **`clear-circuit-class`**

Counters

Use the **counters** command to display statistics describing bandwidth reservation traffic for the traffic classes configured for a PPP interface or Frame Relay circuit.

Syntax: counters

Example: **`counters`**

```
Bandwidth Reservation Counters
Interface 1

Class         Pkt Xmit     Bytes Xmit     Bytes Ovfl
LOCAL                0              0              0
DEFAULT              1             30              0
CLASS 1              1             56              0
CLASS 2              0              0              0

TOTAL                2             86              0
```

Note: The Bytes Ovfl column lists the number of bytes for packets that could not be transmitted because either the maximum queue-length was reached for a priority queue or the packet could not be queued because the priority queue was at the minimum queue length threshold and the packet came from an interface that was running low on receive buffers.

Counters-Circuit-Class

Note: Used only when monitoring Frame Relay.

Use the **counters-circuit-class** command to display statistics for the traffic classes configured for a Frame Relay circuit.

Syntax: counters-circuit-class

Example: **counters-circuit-class**

```
Bandwidth Reservation Circuit Class Counters
Interface 1


Class         Pkt Xmit     Bytes Xmit     Bytes Ovfl
DEFAULT             25           3402             26
CIRCLASS1            1             56              0
CIRCLASS2            0              0              0

TOTAL               26           3458             26
```

Interface

Use the **interface** command to select the serial interface to which bandwidth reservation console commands will be applied. *Bandwidth reservation is supported on routers running the PPP (Point-to-Point Protocol) and Frame Relay interfaces.*

Note: To enter bandwidth reservation commands for a new interface, this command must be entered before using any other bandwidth reservation console commands. If you have exited the bandwidth reservation console prompt (BRS>) and want to return to monitor bandwidth reservation, you must again enter this command first.

To monitor Bandwidth Reservation on a particular interface, at the BRS> console prompt, type the number of the interface. You can then use bandwidth reservation console commands as described in this chapter.

Syntax: interface *interface#*

Example: **interface 1**

Last

Use the **last** command to display the last saved t-class statistics. The t-class statistics are displayed in the same format as they are for the **counters** command.

Syntax: last

Example: `last`

Last-Circuit-Class

Note: Used only when monitoring Frame Relay.

Use the **last-circuit-class** command to display the last saved circuit class statistics. The c-class statistics are displayed in the same format as they are for the **counters-circuit-class** command.

Syntax: last-circuit-class

Example: `last-circuit-class`

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

Chapter 12. Using and Configuring MAC Filtering

This chapter describes how to use medium access control (MAC) for specifying packet filters to be applied to packets during processing. It includes the following sections:

Filters are a set of rules applied to a packet to determine how the packet should be handled during bridging. MAC filtering affects only bridged traffic.

Note: MAC Filtering is allowed on tunnel traffic.

During the filtering process, packets are processed, filtered, or tagged during bridging. The actions are:

- **Processed** – Packets are permitted to pass unaffected through the bridge.
- **Filtered** – Packets are not permitted to pass through the bridge.
- **Tagged** – Packets are allowed to pass through the bridge, but are marked with a number in the range 1 through 64 based on a configurable parameter.

A MAC filter consists of the following three objects:

1. Filter-item – which is a single rule that is applied to the address field or an arbitrary window of data within a packet. The result of applying the rule is either a true (successful match) or false (no match) condition.
2. Filter-list – which contains a list of one or more filter-items.
3. Filter – which contains a set of filter-lists.

MAC Filtering and DLSw Traffic

You can filter incoming LLC traffic for the DLSw network by implementing MAC Filtering; you can filter incoming and outgoing DLSw traffic and you can forward incoming DLSw traffic to the bridge.

To set up a filter for LLC, use the *Bridge Net* number as the interface number for the filter. Determine the Bridge Net number by adding two to the number of interfaces configured for your router. Enter the **list devices** command at the `Config>` prompt, or enter **configuration** at the `+` prompt to see a list of interfaces.

In the following example, the Bridge Net number is 7.

```
Ifc 0 Ethernet                    CSR  81600, CSR2  80C00, vector 94
Ifc 1 WAN X.25                    CSR  81620, CSR2  80D00, vector 93
Ifc 2 WAN X.25                    CSR  81640, CSR2  80E00, vector 92
Ifc 3 WAN PPP                     CSR 381620, CSR2 380D00, vector 125
Ifc 4 WAN Frame Relay             CSR 381640, CSR2 380E00, vector 124
Ifc 5 Token Ring                  CSR 600000, vector 95
```

When you set up a filter for the Bridge Net, for example, the router does not drop frames that match exclusive filters. Instead, it forwards those frames to the bridge.

MAC Filtering Parameters

You can specify some or all of the following parameters to create a filter:

- Source MAC address or destination MAC address
- Data to be matched within the packet
- Mask to be applied to the packet's fields to be filtered
- Interface number
- Input/Output designation
- Include/Exclude/Tag designation
- Tag value (if the tag designation is given)

Filter-Item Parameters

The following parameters are used to construct an address-filter-item:

- Address Type: SOURCE or DESTINATION
- Tag: a *tag-value*
- Address Mask: a *hex-mask*

Each filter-item specifies an address type (either SOURCE or DESTINATION) to match against the type in the packet.

The address mask is a string of numbers entered in hex, which is used in comparing the packet's addresses. The mask is applied to the SOURCE or DESTINATION MAC address of the packet before comparing it against the specified MAC address.

The address mask must be of equal length to the MAC address and specifies the bytes that are to be logically ANDed with the bytes in the MAC address before the equality comparison to the specified MAC address is made. If no mask is specified, it is assumed to be all 1s.

Filter-List Parameters

The following parameters are used to construct a filter-list:

- Name: an *ASCII-string*
- Filter-item list: *filter-item 1* . . . *filter-item n*
- Action: INCLUDE, EXCLUDE, TAG(*n*)

A filter-list is built from one or more filter-items. Each filter-list is given a unique name.

Applying a filter-list to a packet consists of comparing each filter-item in the order in which the filter-items were added to the list. If any filter-item in the list returns a TRUE condition then the filter-list will return its designated action.

Filter Parameters

The following parameters are used to construct a filter:

- Filter-list names: *ASCII-string 1* . . . *ASCII-string n*
- Interface number: an *IFC-number*
- Port direction: INPUT or OUTPUT
- Default action: INCLUDE, EXCLUDE, or TAG
- Default tag: a *tag-value*

A filter is constructed by associating a group of filter-list names with an interface number and assigning an INPUT or OUTPUT designation. The application of a filter to a packet means that each of the associated filter-lists should be applied to packets being received (INPUT) or sent (OUTPUT) on the specified numbered interface.

When a filter evaluates a packet to an INCLUDE condition, the packet is forwarded. When a filter evaluates a packet to an EXCLUDE condition, the packet is dropped. When a filter evaluates to a TAG condition, the packet being considered is forwarded with a tag.

An additional parameter of each filter is the default action, which is the result of non-match for all of its filter-lists. This default action is INCLUDE. It can be set to INCLUDE, EXCLUDE, or TAG. In addition, if the default action is TAG, a tag value is also given.

Using MAC Filtering Tags

The following list includes some uses of MAC filtering tags

- MAC Address filtering is handled jointly by bandwidth reservation and the MAC Filtering feature (MCF) using tags. A user with bandwidth reservation is able to categorize bridge traffic, for example, by assigning a tag to it.
- The tagging process is done by creating a filter-item in the MAC Filtering configuration console and then assigning a tag to it. This tag is then used to set up a bandwidth class for all packets associated with this tag. Tag values must currently be in the range 1 to 64.
- Once a tagged filter has been created in the MAC Filtering configuration process, the Bandwidth Reservation (BRS) **tag** configuration command is used to assign a BRS tag name (TAG1, TAG2, TAG3, TAG4, or TAG5) to the MAC filter tag number. The BRS tag name is then used on the BRS **assign** configuration command to assign the corresponding MAC filter to a bandwidth traffic class and priority.
- Up to 5 tagged MAC addresses can be set from 1 to 5. TAG1 will be searched for first, then TAG2, all the way to TAG5.

:

Tags can also refer to “groups” in IP Tunnel. IP Tunnel end-points can belong to any number of groups, with packets assigned to a particular group through the tagging feature of MAC address filtering.

Accessing the MAC Filtering Configuration Prompt

Use the **feature** command from the CONFIG process to access the MAC filtering configuration commands. The **feature** command lets you access configuration commands for specific features outside the protocol and network interface configuration processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release. For example:

```
Config> feature ?
WRS
BRS
MCF
Feature name or number [MCF]?
```

To access the MAC filtering configuration prompt, enter the **feature** command followed by the *feature number* (3) or *short name* (MCF). For example:

```
Config> feature mcf
MAC Filtering user configuration
Filter config>
```

Once you access the MAC filtering configuration prompt, you can begin entering specific configuration commands. To return to the CONFIG prompt at any time, enter the **exit** command at the MAC filtering configuration prompt.

MAC Filtering Configuration Commands

This section summarizes and explains the MAC filtering configuration commands. Enter these commands at the `Filter config>` prompt.

Use the following commands to configure the MAC filtering feature.

Table 12-1. MAC Filtering Configuration Command Summary

Command	Function
? (Help)	Displays all the MAC filtering commands or lists subcommand options for specific commands (if available).
Attach	Adds a filter list to a filter.
Create	Creates a filter list or an INPUT or OUTPUT filter.
Default	Sets the default action for the specified filter to EXCLUDE, INCLUDE, or TAG.
Delete	Removes all information associated with a filter list. Also deletes a filter that was created using the create filter command.
Detach	Removes a filter list from a filter.
Disable	Disables MAC Filtering entirely or disables a particular filter.
Enable	Enables MAC Filtering entirely or enables a particular filter.
List	Lists a summary of all the filter lists and filters configured by the user. Also generates a list of attached filter lists for this filter and all subsequent information for the filter.
Move	Reorders the filter lists attached to a specified filter.
Reinit	Reinitializes the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.
Set-Cache	Changes the cache size for a filter.
Update	Adds or deletes information from a specific filter list. Brings you to a menu of appropriate subcommands.
Exit	Exits the MAC filtering configuration process.

? (Help)

Use the **?** (help) command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

Attach

Use the **attach** command to add a filter-list to a filter.

A filter is constructed by associating a group of filter-lists with an interface number. A filter-list is built from one or more filter-items.

Syntax: attach *filter-list-name filter-number*

Example: **attach atm_list 3**

Create

Use the **create** command to create a filter-list or an INPUT or OUTPUT filter.

Syntax: create list filter-list-name
filter input/output interface-number

list *filter-list-name*

Creates a filter-list. Lists are named by a unique string (Filter-list-name) of up to 16 characters of the user's choice. This name is used to identify a filter-list that is being built. This name is also used with other commands associated with the filter-list.

Example: **create list newyork**

filter INPUT/OUTPUT *interface-number*

Creates a filter and places it on the network associated with the INPUT or OUTPUT direction on the interface given by an interface number. By default this filter is created with no attached filter-lists, has a default action of INCLUDE and is ENABLED.

Example: **create filter INPUT 2**

Default

Use the **default** command to set the default action for the filter with a specified filter number to exclude, include, or tag.

Syntax: default exclude *filter-number*
include *filter-number*
tag *tag-number filter-number*

exclude *filter-number*

Sets the default action for the filter with a specified filter number to exclude.

Example: **default exclude 3**

include *filter-number*

Sets the default action for the filter with a specified filter number to include.

Example: **default include 3**

tag *tag-number filter-number*

Sets the default action for the filter with the specified filter number to TAG and sets the associated tag value to tag number.

Example: `default tag 3 15`

Delete

Use the **delete** command to remove all information associated with a filter-list and to free an assigned string as a name for a new filter-list. If filter-list is attached to a filter that has already been created by the user, then this command will display an error message on the console without deleting anything. In addition all filter-items belonging to this list are also deleted

This command also deletes a filter that was created using the **create filter** command.

Syntax: delete list *filter-list*
filter *filter-number*

list *filter-list*

Removes all information associated with a filter-list and frees an assigned string as a name for a new filter-list. The filter-list must be a string entered by a previous **create list** command.

If the filter-list is attached to a filter that has already been created by the user, then this command will display an error message on the console without deleting anything. All filter-items belonging to this list are also deleted when this command is used.

Example: `delete list newyork`

filter *filter-number*

Deletes a filter that was created using the **create filter** command.

Example: `delete 3`

Detach

Use the **detach** command to delete a filter-list name (filter-list parameter) from a filter (filter-number parameter).

Syntax: detach *filter-list-name filter-number*

Example: `detach newyork 3`

Disable

Use the **disable** command to disable MAC Filtering entirely or to disable a particular filter.

Syntax: disable all
filter *filter-number*

all

Disables MAC Filtering entirely. Filters are still set as ENABLED, however, if they were enabled previously.

Example: `disable all`

filter *filter-number*

Disables a particular filter. The filter-number parameter corresponds to the numbers displayed in the **list filters** command.

Example: `disable filter 3`

Enable

Use the **enable** command to enable MAC Filtering entirely or to enable a particular filter.

Syntax: enable all
filter *filter-number*

all

Enables MAC Filtering entirely, although filters themselves may still be set to DISABLED.

Example: `enable all`

filter *filter-number*

Enables a particular filter. The filter-number parameter corresponds to the numbers displayed in the **list filters** command.

Example: `enable filter 3`

List

Use the **list** command to list a summary of all the filter-lists and filters configured by the user. A list of all the filter-lists attached to a filter is not given. Other information displayed includes:

- A list containing the state of the filtering system (ENABLE, DISABLE)
- The set of configured filter-list records
- Each of the configured filter records.

In addition, the following information is displayed for each filter:

- Filter number
- Interface number
- Filter direction (INPUT, OUTPUT)
- Filter state (ENABLE, DISABLE)
- Filter default action (TAG, INCLUDE, EXCLUDE).

This command also generates a list of attached filter-lists for this filter and all subsequent information for the filter.

Syntax: list all
filter *filter-number*

Example: `list all`

filter *filter-number*

Generates a list of attached filter-lists for the specified filter and all subsequent information for the filter.

Example: `list filter 3`

Move

Use the **move** command to reorder the filter-lists attached to a specified filter (given by filter-number parameter). The list given by Filter-list-name1 is moved immediately before the list given by Filter-list-name2.

Syntax: move *filter-list-name1 filter-list-name2 filter-number*

Example: **`move newyork boston 13`**

Reinit

Use the **reinit** command to reinitialize the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.

Syntax: reinit

Example: **`reinit`**

Set-Cache

Use the **set-cache** command to change the default cache size (16) to a number in the range 4 to 32768.

Syntax: set-cache *cache-size filter-number*

Example: **`set-cache 32 13`**

Update

Use the **update** command to add information to or delete information from a specific filter-list. Using this command with the desired filter-list-name brings you to the `Filter filter-list-name Config>` prompt for that specific filter-list. From this new prompt you can then change information in the specified list.

The new prompt level is used to add or delete filter-items from filter-lists. The order in which the filter-items are specified for a given filter-list is important as it determines the order in which the filter-items are applied to a packet.

Syntax: update *filter-list-name*

Example: **`update newyork`**

Exit

Use the **exit** command to return to the previous CONFIG prompt level.

Syntax: exit

Example: **`exit`**

Update Subcommands

This section summarizes and then explains the MAC filtering configuration subcommands. Enter these subcommands at the `Filter filter-list-name config>` prompt.

Table 12-2. Update Subcommands Summary	
Subcommand	Function
? (Help)	Displays all the update subcommands.
Add	Adds source or destination MAC address filters or a window filter. Adds filter-items to a filter-list.
Delete	Removes filter-items from a filter-list.
List	Lists a summary of all the filter-lists and filters configured by the user. Also generates a list of attached filter-lists for this filter and all subsequent information for the filter.
Move	Reorders the filter-lists attached to a specified filter.
Set-Action	Sets a filter-item to evaluate the INCLUDE, EXCLUDE or TAG (with a tag-number option) condition.
Exit	Exits the update subcommand configuration process.

Use the following subcommands to update a filter-list.

? (Help)

Use the **? (help)** subcommand to list the commands that are available from the current prompt level. You can also enter a ? after a specific subcommand name to list its options.

Syntax: ?

Example: ?

Add

Use the **add** subcommand to add filter-items to a filter-list. This subcommand specifically lets you add a hexadecimal number to compare against the source or destination MAC address, or a sequence of window data with a mask to compare against a packet data.

The order in which the filter-items are added to a given filter-list is important because it determines the order in which the filter-items are applied to a packet.

Each use of the **add** subcommand creates a filter-item within the filter-list. The first filter-item created is assigned filter-item-number 1, the next one is assigned number 2, and so on. After you enter a successful **add** subcommand, the router displays the number of the filter-item just added.

The first match that occurs stops the application of filter-items, and the filter-list evaluates to INCLUDE, EXCLUDE, or TAG, depending on the designated action of the filter-list. If none of the filter-items of a filter-list produces a match, then the default action (INCLUDE, EXCLUDE or TAG) of the filter is returned.

Syntax: a̲dd source *hex-MAC-addr hex-Mask*
destination *hex-MAC-addr hex-Mask*
window MAC *offset-value hex-data hex-mask*
window INFO *offset-value hex-data hex-mask*

source *hex-MAC-addr hex-Mask*

Adds a hexadecimal number to compare against the source MAC address. ***hex-MAC-addr*** must be an even number of hex digits with a maximum of 16 digits and should be entered without a 0x in front.

The hex-mask parameter must be the same length as hex-MAC-address and is logically ANDed with the designated MAC address in the packet. The default hex-mask argument is to be all binary 1s.

The hex-MAC-addr parameter can be specified in canonical or noncanonical bit order. A canonical bit order is specified as just a hex number (for example, 000003001234). It may also be represented as a series of hex digits with a hyphen (-) between every two digits (for example, 00-00-03-00-12-34).

A noncanonical bit order is specified as a series of hex digits with a colon (:) between every two digits (for example, 00:00:C9:09:66:49). MAC addresses of filter-items will always be displayed using either a hyphen (-) or a colon (:) to distinguish canonical from noncanonical representations.

Example: `add source 00-00-03-00-12-34 00-00-00-00-12-34`

destination *hex-MAC-addr hex-Mask*

Acts identically to the add source subcommand, with the exception that the match is made against the destination rather than the source MAC address of the packet.

Example: `add destination 00-00-03-00-12-34 00-00-00-00-12-34`

window MAC *offset-value hex-data hex-mask*

Adds a sliding window filter-item using the specified offset (computed from the beginning of the frame) that matches the hex data with the mask against packet data.

Example: `add window mac 14 f4f403 ffffff`

window INFO *offset-value hex-data hex-mask*

Similar to the **add window mac** command, except that the offset is computed with respect to the beginning of the information field.

Example: `add window info 0 f4f403 ffffff`

Delete

Use the **delete** subcommand to remove filter-items from a filter-list. You delete filter-items by specifying the filter-item-number assigned to the item when it was added.

When the **delete** subcommand is used, any gap created in the number sequence is filled in. For example, if filter-items 1, 2, 3, and 4 exist and filter-item 3 is deleted, then filter-item 4 will be renumbered to 3.

Syntax: delete *filter-item-number*

Example: `delete 3`

List

Use the **list** subcommand to print out a listing of all the filter-item records. The following information about each MAC-Address filter-item is displayed:

- MAC address and address mask in canonical or noncanonical form.
- filter-item numbers
- address type (source or destination)
- filter-list action

Syntax: list canonical
noncanonical
mac-address canonical
mac-address noncanonical
window

canonical
: Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in canonical form, and the address mask in canonical form. It also gives the filter-list action.

mac-address canonical
: Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in canonical form, and the address mask in canonical form. In addition the filter-list action is given.

  Example: `list canonical list mac-address canonical`

noncanonical
: Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in noncanonical form, and the address mask in noncanonical form. It also gives the filter-list action.

mac-address noncanonical
: Prints out a listing of all the filter-item records within a filter-list, giving the item numbers, the address type (SRC, DST), the MAC address in noncanonical form, and the address mask in noncanonical form. It also gives the filter-list action.

  Example: `list noncanonical list mac-address noncanonical`

window
: Prints out a listing of all the sliding window filter-item records within a filter-list, giving the item numbers, base, offset, data, and mask. It also gives the filter-list action.

  Example: `list window`

Move

The **move** subcommand reorders filter-items within the filter-list. The filter-item whose number is specified by *filter-item-name1* is moved and renumbered to be just before *filter-item-name2*.

Syntax: move *filter-item-name1 filter-item-name2*

Example: `move 2 4`

Set-Action

The **set-action** subcommand lets you set a filter-item to evaluate the INCLUDE, EXCLUDE, or TAG (with a tag-number option) condition. If one of the filter-items of the filter-list matches the contents of the packet being considered for filtering, the filter-list will evaluate to the specified condition. The default setting is INCLUDE.

Syntax: set-action *INCLUDE or EXCLUDE or TAG tag-number*

Example: `set action EXCLUDE`

Exit

Use the **exit** subcommand to return to the previous prompt level.

Syntax: exit

Example: `exit`

Chapter 13. Monitoring MAC Filtering

This chapter describes how to access the MAC Filtering console prompt and how to use the available console commands. It includes the following sections:

- "Accessing the MAC Filtering Console Prompt"
- "MAC Filtering Console Commands"

Accessing the MAC Filtering Console Prompt

Use the **feature** command from the GWCON process to access the MAC filtering console commands. The **feature** command lets you access console commands for specific router features outside of the protocol and network interface console processes.

Enter a question mark after the **feature** command to obtain a listing of the features available for your software release. For example:

```
+ feature ?
WRS
BRS
MCF
```

To access the MAC filtering console prompt, enter the **feature** command followed by the feature number (3) or short name (MCF). For example:

```
+ feature mcf
MAC Filtering user console
Filter>
```

Once you access the MAC filtering console prompt, you can begin entering specific console commands. To return to the GWCON prompt at any time, enter the **exit** command at the MAC Filtering console prompt.

MAC Filtering Console Commands

This section summarizes and explains the MAC filtering console commands. Enter these commands at the `Filter>` prompt.

Table 13-1. MAC Filtering Console Command Summary

Command	Function
? (Help)	Displays all the MAC filtering commands or lists subcommand options for specific commands (if available).
Clear	Clears the "per filter" statistics listed in the list filter command.
Disable	Disables MAC Filtering globally or on a "per filter" basis.
Enable	Enables MAC Filtering globally or on a "per filter" basis.
List	Lists a summary of statistics and settings for each filter currently running in the router.
Reinit	Reinitializes the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.
Exit	Exits the MAC filtering console process.

Use the following commands to monitor the MAC filtering feature.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: ?

Clear

Use the **clear** command to clear filter statistics.

Syntax: clear all
filter *filter-number*

all

Clears the statistics listed by the **list all** command.

Example: `clear all`

filter *filter-number*

Clears the statistics listed by the **list filter** command.

Example: `clear filter 3`

Disable

Use the **disable** command to disable MAC filtering globally. This command does not individually disable each filter.

The command also disables a filter as specified by filter-number. This filter is disabled without modifying configuration records. If no argument is given, MAC filtering is globally disabled.

Syntax: disable all
filter *filter-number*

all

Disables MAC filtering globally. This command does not individually disable each filter.

Example: `disable all`

filter ***filter-number***

Disables the filter that is specified by the filter number. This filter is disabled without modifying configuration records. If no filter number is given, MAC filtering is globally disabled.

Example: `disable filter 3`

Enable

Use the **enable** command to enable MAC filtering globally. This command does not individually enable each filter.

The command also enables a filter as specified by filter-number. This filter is enabled without modifying configuration records. If no argument is given, MAC filtering is globally enabled.

Syntax: enable all
filter *filter-number*

all
Enables MAC filtering globally. This command does not individually enable each filter.

Example: `enable all`

filter *filter-number*
Enables the filter that is specified by the filter number. This filter is enabled without modifying configuration records. If no filter number is given, MAC filtering is globally enabled.

Example: `enable filter 3`

List

Use the **list** command to list a summary of statistics and settings for each filter currently running in the router. The following information is displayed for each filter when the **list all** command is used:

- Default action
- Cache size
- Default tag
- State (enabled/disabled)
- Number of packets which have been filtered as INCLUDE, EXCLUDE or TAG.

In addition, the following information is also displayed by the **list filter** command for a specified filter:

- All information displayed by the list all command
- All the filter-lists currently running in this filter including:
  - List name
  - List action
  - List tag
  - Number of packets which have been filtered by each filter-list.

Syntax: list all
filter *filter-number*

all
Lists statistics and settings for each filter currently running in the router.

Example: `list all`

filter *filter-number*
Generates statistics and settings for each filter plus all the filter-lists currently running in this filter.

Example: `list filter 3`

Reinit

Use the **reinit** command to reinitialize the entire MAC Filtering system from an updated configuration, without affecting the rest of the router.

Syntax: reinit

Example: `reinit`

Exit

Use the **exit** command to return to the previous GWCON prompt level.

Syntax: exit

Example: `exit`

Chapter 14. Configuring WAN Restoral

This chapter includes the following sections:

Overview for WAN Restoral, WAN Reroute, and Dial-on-0verview

The WAN Restoral, WAN Reroute, and Dial-on-overflow features have similar functions and might be confused. This overview is intended to help you decide which of these functions will be useful to you and to help you find the information you need to configure them.

The configuration commands for all three features are included in the "Configuring WAN Restoral" chapter. For additional information about WAN Reroute and Dial-on-overflow see Chapter 16, "The WAN Reroute Feature" on page 16-1.

WAN Restoral

WAN Restoral is the most basic function. When you use WAN Restoral, you configure a primary and a secondary link. In case the primary link fails, the secondary link is started and assumes the characteristics of the primary. You don't configure any protocol definitions on the secondary link because it uses the protocol definitions from the primary link.

For WAN Restoral:

- There is a pairing between a primary and a secondary link.
- You can configure only one primary to use a specific secondary link.
- You don't configure protocol definitions (for example: protocol addresses) on the secondary link.
- The primary link must be a PPP serial interface, it can not be a PPP dial circuit interface.
- The secondary link must be a PPP dial circuit or a Multilink-PPP interface.
- You must enable the wrs feature using the **enable wrs** command.
- You must enable the primary/secondary pair using the **enable secondary-circuit** command.

Note: When BRS is configured on a primary link and the primary link is part of a primary-secondary pair for WAN Restoral, you must configure BRS on the secondary link. Typically when WAN Restoral is configured, the secondary link takes the identify of the primary link. However, this is not true for BRS; therefore, BRS needs to be configured on both the primary and secondary link.

WAN Reroute

WAN Reroute is a more advanced function. When you use WAN Reroute, you configure a primary and an alternate link. In case the primary link fails, the alternate link is started. The routing protocols (for example, RIP or OSPF) detect the newly available link and adjust the routes that are used for forwarding packets.

For WAN Reroute:

- There is a pairing between a primary and an alternate link.
- You may configure multiple primary links to use the same alternate link.
- You must configure protocol definitions on the alternate link.
- The primary link may be any link on which you can configure routable protocols (e.g. IP, IPX). For example, the primary link may be a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The following are examples of interface types that cannot be primary links: SDLC serial interfaces, SRLY serial interfaces, dial-out interfaces, and base nets like V.25bis and ISDN.
- The alternate link may be any link on which you can configure routable protocols (e.g. IP, IPX) and the data-link type of the alternate link need not match the data-link type of the primary link. For example, the alternate link may be a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The following are examples of interface types that cannot be alternate links: SDLC serial interfaces, SRLY serial interfaces, dial-out interfaces, and base nets like V.25bis and ISDN.
- The alternate link may not be a dial-on-demand dial circuit (you must configure 'set idle 0' on the dial circuit).
- You must enable the wrs feature using the **enable wrs** command.
- You must enable the primary/alternate pair using the **enable alternate-circuit** command.
- You may optionally configure stabilization times and start-and stop-time-of-day-revert-back times to control the switching back to the primary link.
- If the alternate link is X.25, you should use the **national-personality set disconnect-procedure active** command when configuring the X.25 interface of the router that has WAN Reroute enabled and use the **national-personality set disconnect-procedure passive** command when configuring the X.25 interface of the other router."

Dial-on-overflow

Dial-on-overflow is similar to WAN Reroute, but does no

t require failure of the primary to start the alternate link. Instead, the utilization of the primary link is monitored, and if a threshold is exceeded, the alternate link is started. Also, not all protocols are brought up on the alternate link. Only IP is brought up on the alternate link, and other protocols continue to use the primary link unless the primary link goes down.

If the primary link goes down, WAN Reroute takes over and any protocols configured on the alternate interface can start detecting and using routes on the alternate interface.

For Dial-on-overflow:

- Dial-on-overflow uses the primary/alternate pairing of a WAN Reroute pair.
- You must configure a WAN reroute pair to use Dial-on-overflow, and all the restrictions of WAN Reroute configuration apply.
- The primary link of a WAN Reroute pair that will be used for Dial-on-overflow must be Frame Relay.
- You must use the OSPF routing protocol to use Dial-on-overflow.
- You must use the **enable dial-on-overflow** command to configure add-threshold and drop-threshold, the bandwidth monitoring interval, and the minimum alternate up time.
- Stabilization times and start-time-of-day-revert-back and stop-time-of-day-revert-back times do not affect the operation of dial-on-overflow.

For more information about WAN Reroute see Chapter 16, “The WAN Reroute Feature” on page 16-1 .

Before You Begin

Before you configure WAN Restoral, you must have the following:

1. A primary serial interface (leased line) configured for PPP. You can use any serial interface on the router.
2. An interface with the associated dial circuits configured on the router. You can use an ISDN interface, a V.25bis interface, or V.34 interface as the base net.
3. A secondary dial circuit configured to dial when the primary interface goes down. To configure a dial circuit to do this, set the idle timer to zero using the **set idle** command at that dial `Circuit Config>`prompt.
4. A secondary dial circuit at one end of the link configured to send calls only. Use the **set calls outbound** command at the `Circuit Config>` prompt.

   Note: Do not configure any protocol addresses on the secondary interface. The protocol assignments for the primary interface are used on the secondary link (dial circuit) when it is active.
5. A secondary dial circuit at the other end of the link configured to receive calls only. Use the **set calls inbound** command at the `Circuit Config>` prompt.

Configuration Procedure for WAN Restoral

This section describes the steps required to configure WAN Restoral. Before you begin, use the **list device** command at the `Config>` prompt to list the interface numbers of different devices.

Follow these steps to configure WAN Restoral on the router:

1. Display the `WRS Config>` prompt by entering the **feature wrs** command at the `Config>` prompt. For example:

```
Config>feature wrs
WAN Restoral user configuration
WRS Config>
```

2. Assign a secondary dial circuit to the primary interface. This dial circuit will back up the primary interface. For example:

```
WRS Config>add secondary-circuit
Secondary interface number  [0]? 3
Primary interface number  [0]? 1
```

3. Enable WAN Restoral on the secondary dial circuit that you added. For example:

```
WRS Config>enable secondary-circuit
Secondary interface number  [0]? 3
```

4. Globally enable WAN Restoral on the router. For example:

```
WRS Config>enable wrs
```

5. Restart the router for configuration changes to take effect.

Secondary Dial Circuit Configuration

To configure a dial circuit:

1. Determine the dial-circuit interface number: To do this, type:

```
Config> list device
```

If no PPP dial-circuit interface is listed, add a dial-circuit interface by typing:

```
Config> add device dial-circuit

Adding device as interface 3
Defaulting Data-link protocol to PPP
Use "net 3" command to configure circuit parameters
```

2. Configure the secondary interface (dial circuit) to have the same data-link type as the primary interface (PPP) from the `Config>` prompt as follows:

```
Config> set data PPP
Interface Number [0]? 3
```

3. Access the dial circuit configuration prompt (`Circuit Config>`) by entering **network** *interface#*.

```
Config> network 3
```

4. Select the base net interface for the dial circuit. The base net can be V.25bis, ISDN, or V.34.

```
Circuit Config> set net 2
```

5. Set the dial circuit idle timer to 0 (0=fixed) as follows:

```
Circuit Config> set idle 0
```

6. Set one end of the backup connection to receive calls (for example, router A) as follows:

```
Circuit Config> set calls inbound
```

7. Set the other end of the backup connection to initiate calls (for example, router B) as follows:

```
Circuit Config> set calls outbound
```

Notes:

1. Do not use the **set calls both** command. Setting these individually will help prevent the collisions of incoming and outgoing connection attempts.
2. Do not configure any forwarder (for example, IP, IPX, etc.) addresses on the dial circuit. The protocol assignments for the primary interface are used on the secondary interface (dial circuit) when it is active.
3. For ISDN configuration instructions, see Chapter 47, “Using and Configuring the ISDN Interface” on page 47-1.
4. For V.25bis configuration instructions, see Chapter 43, “Using and Configuring the V.25bis Network Interface” on page 43-1.
5. For V.34 configuration instructions, see Chapter 45, “Using and Configuring the V.34 Network Interface” on page 45-1.

WAN Restoral, WAN Reroute, and Dial-on-Overflow Configuration Commands

The WAN Restoral configuration commands allow you to create or modify the WAN Restoral interface configuration. This section summarizes and explains the WAN Restoral configuration commands.

Table 14-1 lists the WAN Restoral configuration commands and their function. Enter these commands at the `WRS Config>` prompt. To access `WRS Config>`, enter **feature wrs** at the `Config>` prompt.

Table 14-1. WAN Restoral Configuration Commands Summary

Command	Function
? (Help)	Lists the configuration commands or lists any parameters associated with that command.
Add	Adds a mapping of primary-to-secondary (for WAN Restoral) or primary-to-alternate (for WAN Reroute).
Disable	Disables WRS, an individual secondary-circuit mapping, or alternate-circuit mapping.
Enable	Enables WRS, an individual secondary-circuit mapping, or alternate-circuit mapping.
List	Displays the current Restoral configuration.
Remove	Removes a primary to secondary mapping or a primary to alternate mapping created by add.
Set	Sets the values for the stabilization and time-of-day-revert-back timers.
Exit	Exits the WAN Restoral configuration process.

? (Help)

Use the **?** (help) command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: ?

```
Add
Disable
Enable
List
Remove
Set
Exit
```

Add

Use the **add** command to identify a secondary or an alternate dial-circuit or leased link interface for a primary serial link.

Syntax: add alternate-circuit
secondary-circuit

alternate-circuit

The **add alternate-circuit** command binds an alternate interface to a primary interface for WAN Reroute purposes. You can assign multiple primaries to a single alternate interface. The alternate link type need not be the same as the primary link type (for example, the alternate link type can be a PPP dial circuit and the primary link type can be a Frame Relay leased line).

Example: add alternate-circuit

```
WRS Config>add alt
Alternate interface number  [0]? 6
Primary interface number  [0]? 1
```

Alternate interface number	This is the interface number previously assigned to the alternate interface. Any LAN interface, PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit is an eligible alternate interface. The default is 0.
Primary interface number	This is the interface number of the primary interface previously assigned when the device was added. A primary interface can be any previously defined LAN interface, PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The default is 0.

secondary-circuit

The **add secondary-circuit** command binds a secondary interface to a primary interface for WAN Restoral purposes. Both interfaces must have previously been configured. You can only assign one secondary interface to a primary and vice-versa.

Example: add secondary-circuit

```
WRS Config>add secondary-circuit
Secondary interface number  [0]? 4
Primary interface number  [0]? 1
```

Secondary interface number	This is the dial circuit interface number previously assigned to the secondary interface when the device was added. Any PPP dial circuit or Multilink PPP interface can be a secondary interface. The default is 0.
Primary interface number	This is the interface number of the primary interface previously assigned when the device was added. A primary interface can be any previously defined leased-line running PPP. The default is 0.

Disable

Use the **disable** command to disable the WAN Restoral function, or to disable a primary/secondary pairing for WAN Restoral, or to disable a primary/alternate pairing for WAN Reroute, or to disable Dial-on-overflow for a primary/alternate pairing.

Syntax: disable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit *interface#*

Disables the primary/alternate pairing for WAN Reroute.

Example: **`disable alternate-circuit`**

```
WRS Config> disable alternate-circuit
Alternate interface number  [0]? 6
```

Alternate interface number	This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

dial-on-overflow

Disables dial-on-overflow for all primary/alternate pairings using a specified alternate.

Example: **`disable dial-on-overflow`**

```
WRS Config> disable dial-on-overflow
alternate interface number  [0]? 6
```

Alternate interface number	This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

secondary-circuit *interface#*

Disables the restoral of a particular primary interface by its associated secondary interface until the next **enable secondary-circuit** command at the WRS console. Both interfaces must have been previously configured and bound together in the WRS configuration.

Example: **`disable secondary-circuit`**

```
WRS Config> disable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number	This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs

Disables the WAN Restoral feature globally on the router. This means that WAN Reroute and Dial-on-overflow are also disabled.

Example: **`disable wrs`**

```
WRS Config>  disable wrs
```

Enable

Use the **enable** command to enable the WAN Restoral function, to enable a primary/secondary pairing for WAN Restoral, to enabled a primary/alternate pairing for WAN Reroute, or to enable dial-on-overflow for a primary/alternate pairing.

Syntax: enable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit *interface#*

Enables an alternate circuit

Example: **enable alternate-circuit**

```
WRS Config>enable alternate-circuit
Alternate interface number  [0]? 6
```

Alternate interface number	This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.

dial-on-overflow

Enables dial-on-overflow and allows you to set parameters that control how dial-on-overflow works.

Example: **enable dial-on-overflow**

```
WRS>enable dial-on-overflow

For dial-on-overflow, only IP traffic can overflow to the alternate
interface.
Primary interface number  ]0]? 1
add-threshold (1-100% utilization) [90]?
drop-threshold(0-99% utilization) [60]?
bandwidth test interval(10-200 seconds) [15]?
minimum time to keep the alternate up (20-21600 sec.) [300]?
Dial-on overflow is enabled.
Remember to configure the primary interface's line speed!
```

Primary interface number	This is the interface number of the primary interface for which you are enabling dial-on-overflow. The default is 0.
add-threshold	Determines when an alternate interface will be brought up for additional bandwidth. This value must expressed as a percentage of the primary interface's configured line speed. The default is 90%.
drop-threshold	Determines when an alternate interface is no longer needed for additional bandwidth. This value must expressed as a percentage of the primary interface's configured line speed. The default is 60%.
bandwidth monitoring interval	Determines how often the primary interface's bandwith is monitored for the *add-threshold* and *drop-threshold.* The default is 15 seconds.
Minimum time to keep alternate up	This time period needs to include enough time for the routers to establish the new route when IP traffic on the local router is re-routed to the alternate interface. The default is 5 minutes.

secondary-circuit *interface#*

Enables the restoral of a primary link by the indicated secondary link.

Example: **enable secondary-circuit**

```
WRS Config>enable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number
: This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs
: Enables the function of the WAN Restoral feature on the router. This means that if WAN Reroute and Dial-on-overflow are configured they are also enabled.

Example: **`enable wrs`**

```
WRS Config> enable wrs
```

List

Use the **list** command to display global configuration information for the feature and display configuration information for WAN Restoral primary-secondary pairs, WAN Reroute primary-alternate pairs, and Dial-on-Overflow.

Syntax: list

Example: **`list`**

```
WRS Config>list
WAN Restoral is enabled.
Default Stabilization Time:       0 seconds
Default First Stabilization Time: 0 seconds

                                                    Secondary
   Primary Interface       Secondary Interface      Enabled
-----------------------  -----------------------  --------
 4 - WAN PPP              7 - PPP Dial Circuit       No

                                          Alt.     1st  Subseq TOD Revert Back
Primary Interface  Alternate Interface    Enabled  Stab  Stab  Start    Stop
-----------------  ----------------------  -------  ----  ----  -------  -------

 1 - WAN Frame Re   2 - WAN Frame Relay      Yes    dflt  dflt  Not Set  Not Set


Dial-on-overflow is enabled.
Primary    add-       drop-      test       minimum
Interface  threshold  threshold  interval   alt up time
---------  ---------  ---------  ---------  -----------
     1         29%        20%      15 sec.   300 sec.
```

Remove

Use the **remove** command to delete the mapping of an alternate interface or secondary (backup) interface to the primary interface.

Syntax: remove alternate-circuit
secondary-circuit

Syntax: remove alternate-circuit...

alternate-circuit
: Removes the mapping of a alternate (backup) interface to the primary interface for WAN Reroute. Both interfaces must have been previously assigned and bound together using the **add alternate-circuit** command.

  Example: **`remove alternate-circuit`**

```
WRS Config> remove alternate-circuit
Alternate interface number  [0]? 3
Primary interface number  [0]? 1
```

Alternate interface number	This is the number of the alternate interface previously configured with the **add alternate-circuit** command. The default is 0.
Primary interface number	This is the interface number of the primary interface previously bound to the alternate being removed. The default is 0.

Syntax: remove secondary-circuit...

secondary-circuit

Removes the mapping of a secondary (backup) interface to the primary interface for WAN Restoral. Both interfaces must have been previously assigned and bound together using the **add secondary-circuit** command.

Example: **remove secondary-circuit**

```
WRS Config> remove secondary-circuit
Secondary interface number  [0]? 3
Primary interface number  [0]? 1
```

Secondary interface number	This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.
Primary interface number	This is the interface number of the primary interface previously bound to the secondary being removed. The default is 0.

Set

Use the **set** command to set the parameters for WAN Reroute

Syntax: set ? default
first-stabilization
stabilization
start-time-of-day-revert-back
stop-time-of-day-revert-back

default

Use the **set default** command to set the defaults to be used by links that don't have configured stabilization and first-stabilization times.

Example:

set default

```
WRS Config>set default ?
FIRST-STABILIZATION
STABILIZATION
```

first-stabilization

Sets the default first-stabilization value to be used for links for which a first-stabilization time was not configured.

```
WRS Config>set default first
Default first primary stabilization time (0 - 3600 seconds)  [0]? 20
```

stabilization

Sets the default stabilization value to be used for links for which a stabilization time was not configured.

```
WRS Config>set default stab

Default primary stabilization time (0 - 3600 seconds)  [0]? 30
```

first-stabilization

Sets the number of seconds at router initialization before routing for this primary link is switched to the alternate link if the primary link is not up.

Example:

set first-stabilization

```
WRS Config>set first
Primary interface number  [0]? 1
First primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

First primary stabilization time

The stabilization time for this primary interface. The default is 1.

stabilization

Sets the number of seconds required after the primary link is first detected to be up before routing is switched back to the primary. Routing over the alternate link continues until the primary link remains up for this number of seconds.

Example:

set-stabilization

```
WRS Config>set first
Primary interface number  [0]? 1
Primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting stabilization. The default is 0.

Primary stabilization time

The stabilization time for the primary interface. The default is 1.

start-time-of-day-revert-back

The earliest time of the day the router can switch back to the primary route. The router can revert back to the primary any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary will only occur if the primary is up and the stabilization parameters are met. The default is 0.

Example:

set start-time-of-day-revert-back

```
WRS Config>set start
Primary interface number  [0]? 1
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0] 3
Start time-of-day revert back configured. Remember to configure stop time-of-day
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window start

This time marks the beginning time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the

primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

stop-time-of-day-revert-back

This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

Example:

set stop-time-of-day-revert-back

```
WRS Config>set stop
Primary interface number  [0]? 1
Time-of-Day revert back window stop (1 - 24 hours, 0 = not configured) [0]?5
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window stop

This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

Exit

Use the **exit** command to return to the `Config>` prompt.

Syntax: exit

Example: **`exit`**

Chapter 15. Monitoring WAN Restoral

This chapter describes the WAN-restoral monitoring commands. It includes the following sections:

- "Accessing the WAN Restoral Interface Console Process"
- "WAN Restoral Monitoring Commands"

Accessing the WAN Restoral Interface Console Process

To access the WAN Restoral interface console process, enter the following command at the GWCON (+) prompt:

```
+ feature wrs
```

WAN Restoral Monitoring Commands

The WAN- Restoral (WRS) console commands allow you to monitor the state of WAN- Restoral primary-secondary pairs, WAN Re-route primary-alternate pairs, and Dial-on-Overflow. Any modifications to the operational state of WAN- Restoral, WAN Re-route, and Dial-on-Overflow made through the console interface are not maintained across router restarts.

Access the WRS prompt by entering **feature wrs** at the GWCON (+) prompt. Table 15-1 lists the WRS commands and their functions, and the following sections explain the commands.

Table 15-1. WAN Restoral Monitoring Commands

Command	Function
? (Help)	Lists the monitoring commands or lists any parameters associated with that command.
Clear	Clears the monitoring statistics displayed using the **list** command.
Disable	Disables the WRS, or an individual secondary, or alternate, or dial-on-overflow.
Enable	Enables the WRS, or an individual secondary, or alternate, or dial-on-overflow.
List	Displays the monitoring information on one or all alternate or secondary circuits.
Set	Sets the values for the stabilization and time-of-day-revert-back-timers.
Exit	Returns to the GWCON (+) prompt level.

? (Help)

Use the **?** (help) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: ?

```
Clear
Disable
Enable
List
Set
Exit
```

Clear

Use the **clear** command to clear WAN Restoral, WAN Reroute, and dial-on-overflow statistics that are displayed using the **list** command.

Syntax: clear

Example: **clear**

```
WRS> clear
```

Note: This command clears *Longest restoral period*, but does not clear the *Most recent restoral period*. For the screen display, refer to the example in the **list** command.

Disable

Use the **disable** command to disable the WAN Restoral feature completely, disable the restoral of a particular primary interface by its associated secondary interface, disable an alternate interface or disable dial-on-overflow.

Syntax: disable alternate-circuit
dial-on-overflow
secondary-circuit
wrs

alternate-circuit

Disables a primary/alternate pairing for WAN Reroute. There can be multiple pairings using the same alternate. This command disables all the pairings using the specified alternate-circuit.

Note:

Example: **disable alternate-circuit**

```
WRS>disable alternate-circuit
Alternate circuit number  [0]? 6
```

Alternate circuit number — This is the number of the alternate circuit. The default is 0.

dial-on-overflow

Disables dial-on-overflow for the specified primary/alternate pairing, without changing the enabled/disabled state of WAN Reroute for that pairing. If dial-on-overflow is actively routing, it is terminated at the expiration of the next monitor interval.

Note:

Example: **disable dial-on-overflow**

```
WRS>disable dial-on-overflow
```

secondary-circuit

Disables the restoral of a particular primary interface by its associated secondary interface until the next **restart**, **reload**, or **enable secondary-circuit**

command. Both interfaces must have been previously configured and bound together in the WRS configuration.

Note: Normally, in talk 5 (GWCON), the **disable** command causes the interface to be inactive and stay inactive. For WAN Restoral secondary, however, this is not the case. The **disable** command applied to the secondary interface does not disable the interface itself. It disables only the current call (that is, causes any active call to be disconnected.) To disable use of the secondary circuit, you need to **disable secondary-circuit** at the WAN Restoral console prompt and disable the secondary interface at the top level GWCON prompt.

Example: **`disable secondary-circuit`**

```
WRS>disable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number
: This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs
: Disabling WRS disables WAN Restoral, WAN Reroute, and Dial-on-overflow on the router until the next **restart**, **reload**, or **enable WRS** command.

Example: **`disable wrs`**

```
WRS> disable wrs
```

Enable

Use the **enable** command to enable the WAN Restoral interface, enable the restoral of a primary link by a secondary circuit, enable an alternate circuit, or enable dial-on-overflow.

Syntax: enable
- alternate-circuit
- dial-on-overflow
- secondary-circuit
- wrs

alternate-circuit
: Enables the primary/alternate pairings for WAN Reroute for all pairings using the specified alternate.

Example: **`enable alternate-circuit`**

```
WRS> enable alternate-circuit
Alternate circuit number  [0]? 3
```

Alternate circuit number
: This is the interface number of the alternate circuit. The default is 0.

dial-on-overflow
: Enables dial-on-overflow and allows you to set parameters that control dial-on-overflow. Optionally, allows you to cause the IP protocol to be switched immediately to the alternate, as if the add threshold had been crossed.

Example: **`enable dial-on-overflow`**

```
WRS> dial-on-overflow

For dial-on-overflow, only IP traffic can overflow to the alternate interface.
Primary interface number  [0]? 1
add-threshold (1-100% utilization) [90]?
drop-threshold(0-99% utilization) [60]?
bandwidth test interval(10-200 seconds) [15]?
minimum time to keep the alternate up (20-21600 sec.) [300]?
Dial-on overflow is enabled.
Remember to configure the primary interface's line speed!

Do you want to switch IP traffic to the alternate now?(Yes or [No]):
WRS>
```

secondary-circuit

Enables the restoral of a primary link by the indicated secondary link.

Example: **enable secondary-circuit**

```
WRS> enable secondary-circuit
Secondary interface number  [0]? 3
```

Secondary interface number — This is the number of the secondary interface previously configured with the **add secondary-circuit** command. The default is 0.

wrs

Enables the function of the WAN Restoral feature on the router. This feature needs to be enabled in order to do WAN Restoral, WAN Reroute, or Dial-on-overflow.

Example: **enable wrs**

```
WRS> enable wrs
```

Set

Use the **set** command to set the parameters for WAN Reroute.

Syntax: set ? default
first-stabilization
stabilization
start-time-of-day-revert-back
stop-time-of-day-revert-back

default

Use the **set default** command to set the defaults to be used by links that don't have configured stabilization and first-stabilization times.

Example:

set default

```
WRS Config>set default ?
FIRST-STABILIZATION
STABILIZATION
```

first-stabilization

Sets the default first-stabilization value to be used for links for which a first-stabilization time was not configured.

```
WRS Config>set default first
Default first primary stabilization time (0 - 3600 seconds)  [0]? 20
```

stabilization

Sets the default stabilization value to be used for links for which a stabilization time was not configured.

```
WRS Config>set default stab

Default primary stabilization time (0 - 3600 seconds)  [0]? 30
```

first-stabilization

Sets the number of seconds at router initialization before routing for this primary link is switched to the alternate link if the primary link is not up.

Example:

set first-stabilization

```
WRS Config>set first
Primary interface number  [0]? 1
First primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

First primary stabilization time

The stabilization time for this primary interface. The default is 1.

stabilization

Sets the number of seconds required after the primary link is first detected to be up before routing is switched back to the primary. Routing over the alternate link continues until the primary link remains up for this number of seconds.

Example:

set-stabilization

```
WRS Config>set first
Primary interface number  [0]? 1
Primary stabilization time (0 - 3600 seconds -1 = default)  [-1]?
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting stabilization. The default is 0.

Primary stabilization time

The stabilization time for the primary interface. The default is 1.

start-time-of-day-revert-back

The earliest time of the day the router can switch back to the primary route. The router can revert back to the primary any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary will only occur if the primary is up and the stabilization parameters are met. The default is 0.

Example:

set start-time-of-day-revert-back

```
WRS Config>set startPrimary interface number  [0]? 1
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0] 3
Start time-of-day revert back configured. Remember to configure stop time-of-day
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window start

This time marks the beginning time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and

the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

stop-time-of-day-revert-back

This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

Example:

set stop-time-of-day-revert-back

```
WRS Config>set stopPrimary interface number  [0]? 1
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0]?
5
```

Primary interface number

This is the primary interface number of the primary interface for which you are setting first-stabilization. The default is 0.

Time-of-day-revert-back-window stop

This time marks the ending time for the revert back window. The router can revert back to the primary interface any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary interface will only occur if the primary interface is up and the stabilization parameters are met. The default is 1.

List

Use the **list** command to display monitoring information on one or all WAN Restoral primary-secondary pairs or one or all WAN Reroute primary-alternate pairs. .

Syntax: list all
alternate-circuit
secondary-circuit
summary

all

Provides summary information, followed by the specific information, for each secondary interface.

Example: **list all**

```
WAN Restoral/Re-route is enabled with 2 circuits configured
    Total restoral attempts =          7 completions   =          7
    Total packets forwarded =         39
    Longest completed restoral period in hrs:min:sec     0:03:27

    Total overflow attempts =         20 completions   =         19
    Longest completed overflow period in hrs:min:sec     0:05:00

   Primary       Secondary    Restoral  Restoral   Current/Longest
Net Interface  Net Interface  Enabled    Active       Duration
-------------  -------------  --------  --------   --------------------
  4  PPP/0       7  PPP/1       No         No       00:03:27/ 00.06.00

                                Re-route/ Re-route/         Recent
   Primary       Alternate    Overflow  Overflow     Reroute/Overflow
Net Interface  Net Interface  Enabled    Active          Duration
-------------  -------------  --------  --------   --------------------
  1   FR/0       2   FR/1     Yes/Yes    No /No     00:00:56/ 00:05:00
```

Total restoral attempts	The number of times the primary link failed, causing the router to try to bring up a secondary link.
Completions	The number of successful restoral attempts when the secondary link came up and was used.
Total packets forwarded	The total number of packets forwarded across the secondary interface. It is the sum of both directions, and is cumulative over all successful restores, until the restart or clear restoral-statistics command is issued.
Longest Completed Restoral Period	This field displays in hours, minutes, and seconds the longest amount of time a restoral was in operation, not counting any current usage.
Total Overflow Attempts	The number of attempts due to an overflow.
Completions	The number of successful overflow attempts when the secondary link came up and was used.
Longest Completed Overflow Period	Displays in hours, minutes , and seconds the longest amount of time an overflow was in operation, not counting any current usage.
Primary Net Interface	The interface that is being backed up by its associated secondary interface.
Secondary Net Interface	The dial circuit that is being used to back up the associated primary interface.
Restoral Enabled	Indicates that restoral of this primary interface is currently enabled.
Restoral Active	Indicates whether restoral is active (Yes or No).
Current/Longest Duration	Indicates in hours, minutes, and seconds the current and longest duration the secondary net interface was up.
Primary Net Interface	The interface that is being backed up by its associated alternate interface.
Alternate Net Interface	The interface that is being used as an alternate back up the associated primary interface.
Re-route/Overflow Enabled	Indicates whether re-route and overflow are enabled (Yes or No).
Re-route/Overflow Active	Indicates whether re-route and overflow are active (Yes or No).
Recent Re-route Overflow Duration	Indicates in hours, minutes, and seconds the recent re-route and overflow duration of the alternate net interface.

Alternate-circuit

Provides totals for an alternate circuit. Allows the console operator to retrieve the WAN Reroute state and associated statistics for each alternate interface and its associated primary mapping.

Example: **`list alternate-circuit`**

```
WRS>li alt 7
Primary    1:FR/0 Frame Relay SCC Serial Line
Alternate  7:PPP/1 Point to Point V.25bis Dial Circuit
reroute Enabled, currently inactive
overflow Enabled, currently inactive
Primary first stabilization time: default (0 seconds)
Primary stabilization time: default (0 seconds)
Time-of-day revert back not configured: start = 0, stop = 0
Restored 0 times (0 attempts)
Overflow 0 times (0 attempts)
```

Primary Interface	The interface that is being backed up by this associated alternate interface.
Alternate Interface	The dial circuit that is being used to back up the associated primary interface.
Reroute Enabled	Indicates whether reroute of this primary interface is currently enabled.
Overflow Enabled	Indicates whether overflow of this primary interface is currently enabled.
Primary first stabilization	The number of seconds at router initialization before routing for this primary link is switched to the alternate link if the primary link is not up.
First stabilization	The number of seconds required after the primary link is first detected to be up before routing is switched back to the primary. Routing over the alternate link continues until the primary link remains up for this number of seconds.
Time-of-day revert back	The time of the day the router can switch back to the primary route. The router can revert back to the primary any time between the start-time-of-day-revert-back time and the stop-time-of-day-revert-back time. Reverting back to the primary will only occur if the primary is up and the stabilization parameters are met. The default is 0.
Restored times	The number of attempts to reroute the primary interface.
Overflow times	The number of dial-on-overflow attempts.

secondary-circuit

Provides totals for each secondary circuit. Allows the console operator to retrieve the WAN Restoral state and associated statistics for each secondary interface and its associated primary mapping.

Example: **`list secondary-circuit`**

```
Secondary interface number  [0]? 1

                                                          Secondary
Primary Interface          Secondary Interface            Enabled
-------------------        --------------------           ---------
1  PPP/0  Point to Poi     3  PPP/1  Point to Poi           Yes

Router primary interface state = Up
Router secondary interface state = Available
Restoral Statistics:

  Primary restoral attempts =       6   completions =    5
  Restoral packets forwarded =    346
    Most recent restoral period in hrs:min:sec          00:08:20
```

Primary Interface	The interface that is being backed up by this associated secondary interface.
Secondary Interface	The dial circuit that is being used to back up the associated primary interface.
Secondary Enabled	Indicates whether restoral of this primary interface is currently enabled.

Router Primary Interface State	Indicates that the primary interface state is one of the following: Up - Indicates that the link is up. Down - Indicates that the link is down. Disabled - Indicates that the operator has disabled the link. Not present - Indicates that the link is configured but there is a hardware problem.
Router Secondary Interface State	Indicates that the associated secondary interface state is one of the following: Up - Indicates that the link is up. Down - Indicates that the link is down. This also occurs when the base network for the secondary is disabled either at the `Config>` prompt or at the operator console. Available - Indicates that the link is in the waiting mode. Testing - Indicates that the link is in the process of establishing a connection.
Restoral Statistics:	
Primary Restoral Attempts	The number of times the primary failed, causing the router to try to bring up a secondary link.
Restoral Packets forwarded	This field indicates the total number of packets forwarded.
Most Recent Restoral Period	This indicates how long the secondary was up, the last time it was used or during the current restoral use.

summary

Provides totals for each secondary circuit.

Example: **`list summary`**

```
WAN Restoral is enabled with 3 circuit(s) configured

  Total restoral attempts =      3 completions =      2
  Total packets forwarded =    346
  Longest restoral period in hrs:min:sec   00:08:20

Primary Interface and State      Secondary Interface and State
---------------------------   ---------------------------------
 1  PPP/0  - Up                 3  PPP/1  - Available
```

Total restoral attempts	The number of times the primary failed, causing the router to try to bring up a secondary link.
Completions	The number of successful restoral attempts when the secondary came up and was used.
Total packets forwarded	The total number of packets forwarded across the secondary interface. It is the sum of both directions, and is cumulative over all restoral periods until the restart or clear restoral-statistics command is used.
Longest restoral period	This field displays in hours, minutes, seconds the longest amount of time restoral was in use, not counting the current usage.

Primary Interface and State	The interface that is being backed up by its associated secondary. Valid states are: Up - Indicates that the link is up. Down - Indicates that the link is down. Disabled - Indicates that the operator has disabled the link. Not present - Indicates that the link is configured but there is a hardware problem.
Secondary Interface and State	The dial circuit that is being used to back up the associated primary. Valid states are: Up - Indicates that the link is up. Down - Indicates that the link is down. This also occurs when the base network for the secondary is disabled either at the `Config>` prompt or at the operator console. Testing - Indicates that the link is in the process of establishing a connection. Available - Indicates that the link is in the waiting mode.

Exit

Use the **exit** command to return to the GWCON (+) prompt.

Syntax: exit

Example: `exit`

Chapter 16. The WAN Reroute Feature

This chapter describes the WAN reroute feature. It includes the following sections:

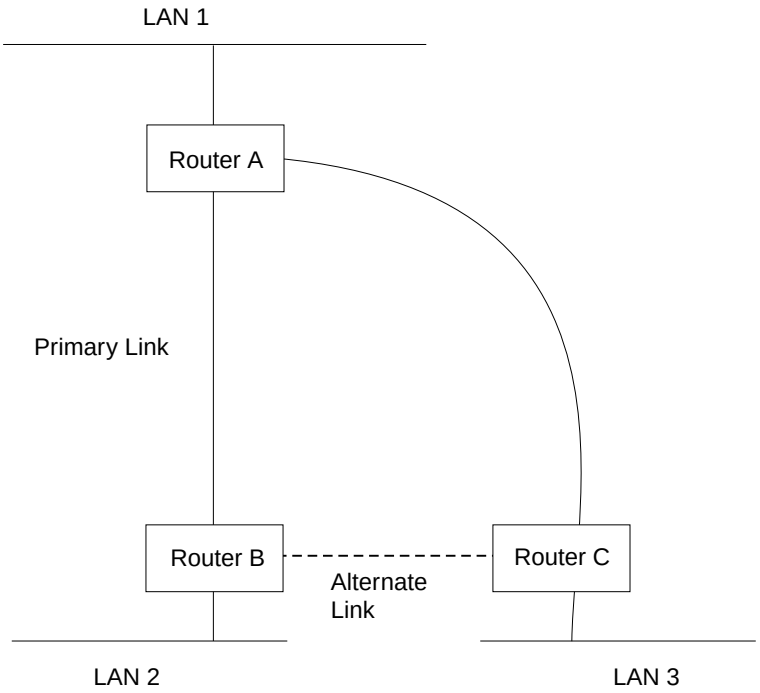
WAN Reroute Overview

WAN Reroute lets you set up an alternate route so that if a primary link fails, the router automatically initiates a new connection to the destination through the alternate route. See "Overview for WAN Restoral, WAN Reroute, and Dial-on-0verview" on page 14-1 for an explanation of WAN Restoral, and how WAN Reroute and Dial-on-overflow work together.

The WAN Reroute process involves:

1. Detecting the primary link failure
2. Switching to the alternate link
3. Detecting the primary link recovery
4. Switching back to the primary link

The alternate link may be any link on which you can configure routable protocols (e.g. IP, IPX) and the data-link type of the alternate link need not match the data-link type of the primary link. For example, the alternate link may be a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit. The following are examples of interface types that cannot be alternate links: SDLC serial interfaces, SRLY serial interfaces, dial-out interfaces, and base nets like V.25bis and ISDN.



If the primary link between routers A and B fails, WAN reroute establishes an alternate link between routers B and C. Routers A and B can then communicate through router C.

Figure 16-1. WAN Reroute. Normally, there is a connection between Routers A and B and Routers A and C.

Dial-on-Overflow

Dial-on-overflow allows you to use an alternate interface for IP traffic when the traffic rate on the primary link reaches a specified threshold. This means that the primary interface does not have to be down before the alternate link is brought up. When the primary interface's traffic reaches the specified threshold the router brings up the alternate link. To use dial-on-overflow, WAN Reroute must be configured and the primary interface must be Frame Relay. IP is the only protocol that can be switched over to the alternate interface by dial-on-overflow. Also, OSPF should be used as the IP routing protocol instead of RIP when dial-on-overflow is used.

For information about configuring dial-on-overflow, see “WAN Restoral, WAN Reroute, and Dial-on-Overflow Configuration Commands” on page 14-5.

Bandwidth Monitoring

The interval for bandwidth monitoring can be specified for dial-on-overflow during WAN Reroute configuration. The primary interface's receive and transmit bandwidth utilization are monitored. When the primary interface's bandwidth reaches the *add* threshold, a WAN Reroute request is generated to bring up the alternate interface. If WAN Reroute is successful bringing up the alternate interface, IP stops routing over the primary interface and starts routing over the alternate interface.

If WAN Reroute is not successful in bringing up the alternate route it periodically attempts to bring up the alternate interface until the primary interface's bandwidth utilization drops below the *drop* threshold.

When the primary interface's receive and transmit bandwidth utilization reaches the *drop* threshold and the minimum configured up time has expired the alternate interface is dropped. This causes IP to stop routing over the alternate interface and start using the primary interface.

The add-threshold and the drop-threshold are specified as a percentage of the configured line speed for the primary link. The configured line speed does not always match the actual speed of the link. The amount of traffic on the link in each direction is calculated separately. The threshold is exceeded if the traffic in either direction is greater then the specified percentage.

Configuring WAN Reroute

Following are the steps required to configure WAN reroute. The next section shows an example of how to perform these tasks.

To configure WAN Reroute, you need to:

1. Configure the primary link.
2. Configure the alternate link.
3. Assign the alternate link to the primary link. You can also specify a stabilization period for the primary link.

   You can specify a time-of-day revert-back to the primary link which will happen after the stabilization period is over (if configured). This allows the secondary to stay up until such time that the user desires and revert back to the primary during off-peak hours.

   Note: The primary and alternate links can be different data-link types. The primary and alternate links can be:

   - A LAN interface.
   - A PPP serial interface.
   - A Frame Relay serial interface.
   - An X.25 serial interface.
   - A PPP dial circuit.
   - A Frame Relay dial circuit.

Sample WAN Reroute Configuration

Figure 16-2 on page 16-4 shows WAN reroute using a Frame Relay dial circuit over ISDN as the alternate link. If the Frame Relay DLCI between router A and router C fails, WAN reroute uses the dial circuit to establish an alternate connection through router D. If one of the primary links from a branch to headquarters fails, WAN reroute establishes an alternate route to headquarters through another branch.

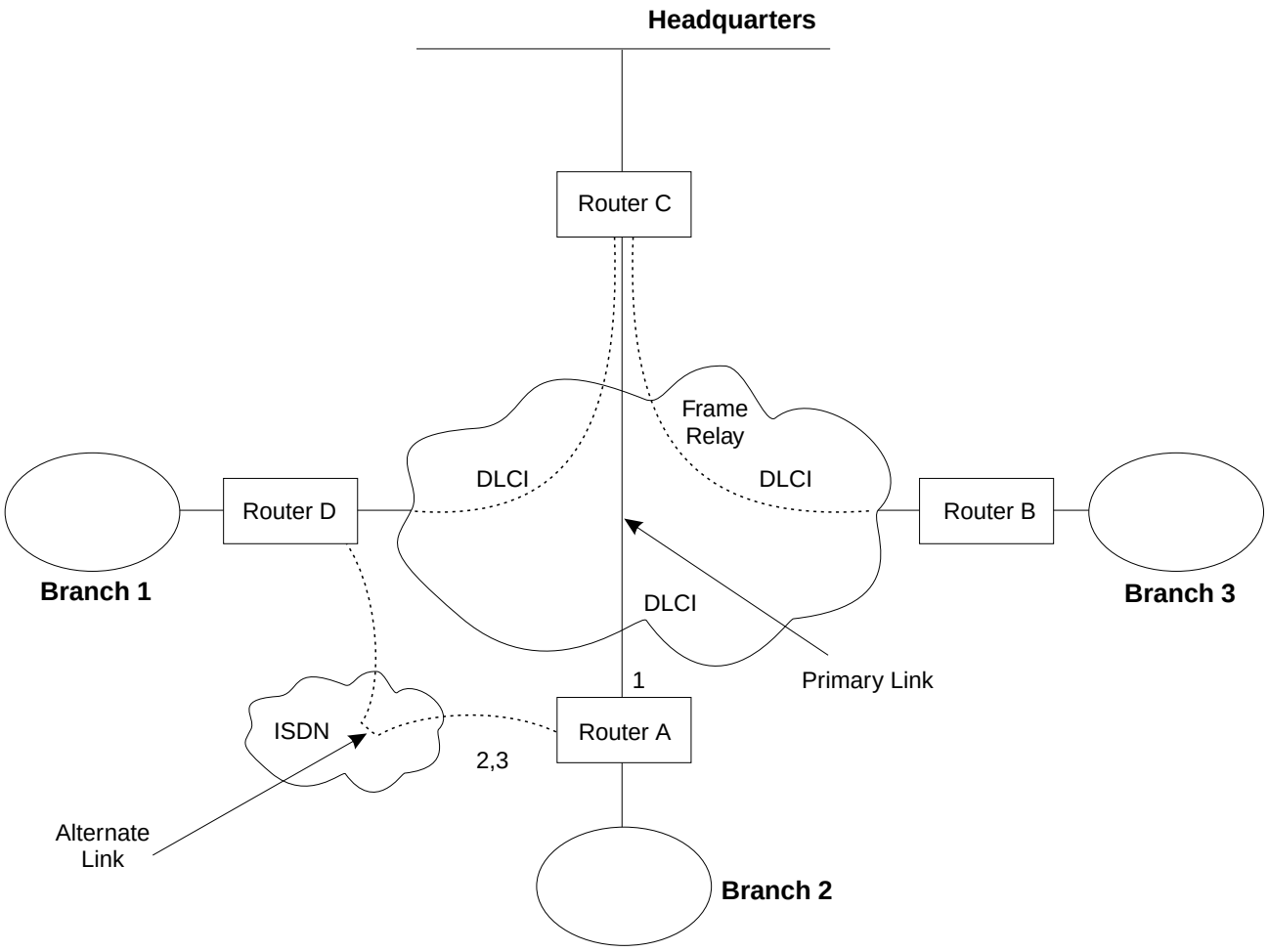


Figure 16-2. Sample WAN Reroute Configuration. Branch offices use frame relay to connect to headquarters.

The following sections describe how to set up WAN reroute on Router A in Figure 16-2. You will need to:

- Configure the primary frame relay interface (1) to have a Required PVC or Required PVC Group or enable the No-PVC feature on the frame relay interface.
- Configure the ISDN interface (2) and its frame relay dial circuit (3).
- Assign the dial circuit to be the alternate link for the primary frame relay interface.
  - Optionally, you can assign:
  - Stabilization period for the primary link,
  - Time-of-day revert-back window for the primary link.

These tasks are described in detail below.

Configuring the Frame Relay Interface

To configure the frame relay interface for WAN reroute, on Router A, add a PVC between Routers A and C on the primary Frame Relay interface.

To cause the primary FR interface to declare itself down when the connection to other router(s) is lost, you have three options:

1. Enable the No-PVC feature. When this feature is enabled, the FR interface goes down when there are no active PVCs.
2. Configure a PVC as required but don't include the PVC in a required PVC group. In this case, the FR interface goes down when the PVC becomes inactive.
3. Configure a set of PVCs as required and as part of a required PVC group. In this case, the FR interface goes down when all of the PVCs of a required PVC group become inactive.

Follow these steps to configure the primary frame relay interface:

1. If you have not yet done so, set the data link on the interface to frame relay.

```
Config>set data-link frame relay
Interface Number [0]? 2
```

2. Enter the Frame Relay configuration process.

```
Config>network
What is the network number [0]?2
Frame Relay user configuration
FR Config>
```

Note: Complete only *one* of the two remaining steps for configuring the primary frame relay interface.

3. Add a PVC using the **add permanent-virtual-circuit** command.

   To configure the PVC as Required:

   Enter **y** to the question “Is circuit required for interface operation ?.”

   To configure the PVC as a member of a required PVC group:

   a. Enter **y** to the question “Does circuit belong to a Required PVC group ?.”
   b. Enter a group name in response to the question “What is the group name ?.”

   If you have already added PVCs, use the **change permanent-virtual-circuit** command to configure the PVC as Required and to assign it to a Required PVC Group, as appropriate. Refer to Chapter 31, “Using and Configuring Frame Relay Interfaces” on page 31-1 for more information.

```
FR Config>add permanent-virtual-circuit
Circuit number [16]?
Committed Information Rate (CIR) in bps [64000]?
Committed Burst Size (Bc) in bits [64000]?
Excess Burst Size (Be) in bits [0]?
Assign circuit name []?
Is circuit required for interface operation [N]?y
Does the circuit belong to a required PVC group [N]? y
What is the group name []?group1
```

4. If desired, enable the No-PVC feature.

   Note: Complete this step *only* if you bypassed the previous step.

```
FR Config>enable no-pvc
```

There are additional parameters that you can set for frame relay. For more information, see Chapter 31, “Using and Configuring Frame Relay Interfaces” on page 31-1.

Configuring the ISDN Interface and Dial Circuit

Configure the ISDN interface and dial circuit between Router A and Router D. See Chapter 47, “Using and Configuring the ISDN Interface” on page 47-1 for information on how to configure ISDN interfaces and dial circuits.

Unlike WAN restoral, you must configure routable protocols on the dial circuit that will be used as the alternate link. If those routable protocols cannot be prevented from sending maintenance packets, the alternate link will establish a connection even if rerouting is not necessary. In this case if you want to use the alternate link only for rerouting, disable the dial circuit. To disable the dial circuit, enter the **disable interface** command at the `Config>` prompt.

If you have multiple dial circuits assigned to the ISDN interface, you can set a priority for the dial circuits. If all the B channels have active dial circuits on the physical interface and a circuit with a higher priority receives a packet, the lowest priority connection is terminated and the high priority circuit establishes a connection.

You can set the priority to between 0 and 15, where 15 is the highest priority circuit and 0 is the lowest priority circuit. The default priority for new dial circuits is 8. Enter **set priority** at the `Circuit Config>` prompt to change the priority.

Assigning and Configuring the Alternate Link

Enter the WAN reroute configuration process to assign the dial circuit as the alternate link for a LAN interface, a PPP, Frame Relay, or X.25 serial interface, or a PPP or Frame Relay dial circuit, and if desired, to specify the stabilization periods and/or the time-of-day revert-back window.

There are two types of stabilization periods:

- *First stabilization period* is the amount of time the router waits for the primary interface to become active when the router first attempts to bring it up. If, after the first stabilization period, the primary has not come up, WAN reroute brings up the alternate link.
- *Stabilization period* is the amount of time the router waits to be sure the primary link is reliable before it switches from the alternate link back to the primary link.

The time-of-day revert-back window is the specific time of day when the user desires the switch back to the primary after it is up and any configured stability time has passed.

Using a 24-hour clock, the user specifies the start and stop hours of the revert back window. The secondary stays up and is not taken down until the start hour is reached. If the time of day when the primary comes up is between the start and stop hours (in the window) then the switch to the primary link is immediate after the stability time is up.

Follow these steps to assign and configure the alternate link:

1. Enter the WAN restoral configuration process.

```
Config>feature wrs
WAN Restoral user configuration
```

2. Assign the dial circuit as the alternate link for the primary frame relay interface.

```
WRS Config>add alternate-circuit
Alternate interface number  [0]? 4
Primary interface number  [0]? 1
```

3. Enable the alternate circuit.

```
WRS Config>enable alternate-circuit
Alternate interface number  [0]? 4
```

4. Optionally, specify a first stabilization period.

   To set the first stabilization period for a specific primary interface, use the **set first-stabilization-period** command. To set a default first stabilization period for all interfaces that do not have specific periods set, use the **set default first-stabilization-period** command.

```
WRS Config>set first-stabilization-period
Primary interface number [0]?
First primary stabilization time (0 - 3600 seconds -1=default) [-1]?

WRS Config>set default first-stabilization-period
Default first primary stabilization time (0 - 3600 seconds) [0]?
```

5. Optionally, specify a stabilization period. To set a stabilization period for specific interfaces use the **set stabilization-period** command. To set a default stabilization period for all interfaces that do not have specific periods set, use the **set default stabilization-period** command.

```
WRS Config>set stabilization-period
Primary interface number [0]?
First primary stabilization time (0 - 3600 seconds -1=default) [-1]?
WRS Config>set default stabilization-period
Default first primary stabilization time (0 - 3600 seconds) [0]?
```

6. Optionally, specify a time-of-day revert-back window.

   To set the start and stop times for specific interface windows use the set start-time-of-day-revert-back and stop-time-of-day-revert-back commands. The default value of zero means no window is configured. The 24-hour clock starts at 1am and ends at 24 midnight. If the start and stop times are the same (but not zero) then the revert back will happen at exactly that hour.

   Following are two examples of setting the revert-back window:

   a. A start time of 23 and a stop time of 3 will give a revert-back window from 11pm until 3am.

   b. A start time of 1 and a stop time of 5 will give a revert-back window from 1am to 5am.

```
WRS Config> set start-time-of-day-revert-back
Primary interface number [0]?
Time-of-Day revert back window start (1 - 24 hours, 0 = not configured) [0]?
WRS Config> set stop-time-of-day-revert-back
Primary interface number [0]?
Time-of-Day revert back window stop (1 - 24 hours, 0 = not configured) [0]?
```

Chapter 17. Using and Configuring the Network Dispatcher Feature

This chapter describes how to configure the Network Dispatcher Feature and contains the following sections:

For additional information about Network Dispatcher, see *Interactive Network Dispatcher User's Guide, GC31-8496*.

Overview of Network Dispatcher

Network Dispatcher is a feature that boosts the performance of servers by routing TCP/IP session requests to different servers within a group of servers, thus balancing the requests among all servers. The routing is transparent to the users and other applications. Network Dispatcher is useful for applications such as e-mail, servers, World Wide Web servers, distributed parallel database queries, and other TCP/IP applications.

Network Dispatcher can help maximize the potential of your site by providing a powerful, flexible, and scalable solution to peak-demand problems. During peak demand periods, Network Dispatcher can automatically find the optimal server to handle incoming requests.

The Network Dispatcher function does not use a domain name server for routing. It balances traffic among your servers through a unique combination of router and management software. Network Dispatcher can also detect a failed server and route traffic around it.

All client requests sent to the Network Dispatcher machine are routed to the server that is selected by the Network Dispatcher as the optimal server according to certain dynamically set weights. You can use the default values for those weights or change the values during the configuration process.

The server sends a response back to the client without any involvement of Network Dispatcher. No additional software is required on your servers to communicate with Network Dispatcher.

The Network Dispatcher function is the key to stable, efficient management of a large, scalable network of servers. With Network Dispatcher, you can link many individual servers into what appears to be a single, virtual server. Your site thus appears as a single IP address to the world. Network Dispatcher functions independently of a domain name server; all requests are sent to the IP address of the Network Dispatcher machine.

Network Dispatcher brings distinct advantages in routing and balancing traffic load to clustered servers, resulting in stable and efficient management of your site.

Balancing TCP/IP Traffic Using Network Dispatcher

There are many different approaches to balancing the load across servers. Some allow users to choose a different server at random if the first server is slow or not responding. Another method is round-robin, in which the domain name server selects a server to handle requests. This approach is better, but does not take into consideration the current load on the target server or even whether the target server is available.

Network Dispatcher can route requests to different servers based on the type of request, an analysis of the load on servers, or a configurable set of weights that you assign. To manage each different type of balancing, the Network Dispatcher has the following components:

Executor Routes connections based on the type of request received. Typical requests types are HTTP, FTP, and SSL. This component always runs.

Advisors Queries the servers and analyzes the results by protocol for each server. The advisor passes this information to the ***manager*** to set the appropriate weight. The advisor is an optional component.

Manager Sets weights for a server based on:

- Internal counters in the executor
- Feedback from the servers provided by the advisors
- Feedback from any system monitoring program

The manager is an optional component. However, if you do not use the manager, the Network Dispatcher will balance the load using a round-robin scheduling method based on the current server weights.

High Availability for Network Dispatcher

The base function Network Dispatcher has the following characteristics that makes it a single point of failure from many different perspectives:

- It examines all the traffic on the way in. If some of the packets for an existing connection use a different path through a different Network Dispatcher to reach a server, the server immediately resets the connection.
- It keeps track of all established connections and although it does not terminate them, entries lost from the Network Dispatcher connection table will result in the resetting of a connection.
- It appears to any previous hop router as the last hop, and the connection's termination.

All these characteristics make the following failures critical for the whole cluster:

- If the Network Dispatcher fails for any reason, all the connection tables are lost, therefore all existing connections from the client to the server are also lost. Assuming there is a second Network Dispatcher that can direct a client to the servers, new connections will be able to go through only after the usual routing protocol delays which could be several minutes.
- If the configured Network Dispatcher interface to the previous IP router fails, there must either be another interface to get to the same Network Dispatcher,

in which case recovery is performed by the IP router (using the ARP aging mechanism with delays in the order of several minutes), or all connections will be lost.

- If Network Dispatcher interface to the servers fails, the previous hop router assumes that the Network Dispatcher is the last hop, and therefore will not reroute new connections. Existing connections will be lost and new connections will not be established.

In all these failure cases, which are not only Network Dispatcher failures but also Network Dispatcher neighborhood failures, all the existing connections are lost. Even with a backup Network Dispatcher running standard IP recovery mechanisms, recovery is, at best, slow and applies only to new connections. In the worst case, there is no recovery of the connections.

To improve Network Dispatcher availability, the Network Dispatcher High Availability function uses the following mechanisms:

- Two Network Dispatchers with connectivity to the same clients, and the same cluster of servers, as well as connectivity between the Network Dispatchers.
- A “keep alive” mechanism between the two Network Dispatchers to detect Network Dispatcher failure.
- A reachability collection, to identify which IP host can and what cannot be reached from each Network Dispatcher.
- Synchronization of the Network Dispatcher databases (that is, the connection tables, reachability tables, and other databases).
- Logic to elect the active Network Dispatcher, which is in charge of a given cluster of servers, and the standby Network Dispatcher, which continuously gets synchronized for that cluster of servers.
- A mechanism to perform fast IP takeover, when the logic or an operator decides to switch active and standby.

Failure Detection

Besides the basic criteria of failure detection, (the loss of connectivity between active and standby Network Dispatchers, detected through the Keepalive messages) there is another failure detection mechanism named “reachability criteria.” When you configure the Network Dispatcher, you provide a list of hosts that each of the Network Dispatchers should be able to reach to order work correctly.

The hosts could be routers, IP servers or other types of hosts. Host reachability is obtained by pinging the host. Switchover takes place either if the Keepalive messages cannot go through, or if the reachability criteria are no longer met by the active Network Dispatcher and the standby Network Dispatcher is reachable. To make the decision based on all available information, the active Network Dispatcher regularly sends the standby Network Dispatcher its reachability capabilities. The standby Network Dispatcher then compares the capabilities with its own and decides whether to switch.

Cache Synchronization

The main data synchronized by the Network Dispatchers are the connection table entries. The Network Dispatcher High Availability function uses a cache synchronization protocol that insures that both Network Dispatchers contain the same entries. This synchronization takes into account a known error margin of transmission delays. The protocol performs an initial synchronization of peer databases and later, maintains the databases through periodic updates.

Recovery Strategy

In the case of a Network Dispatcher failure, the IP takeover mechanism will promptly direct all traffic toward the standby Network Dispatcher. The Database Synchronization mechanism insures that the standby has the same entries as the active Network Dispatcher. When the failure occurs in the network (any intermediate piece of hardware or software between the client and the back-end server), and there is an alternate path through the standby Network Dispatcher that works, the switchover is performed across the alternate path.

IP Takeover

Note: Cluster IP Addresses are assumed to be on the same logical subnet as the previous hop router (IP router).

The IP Router will resolve the cluster address through the ARP protocol. To perform the IP takeover, the Network Dispatcher (standby becoming active) will issue an ARP request to itself, that is broadcasted to all directly attached networks belonging to the logical subnet of the cluster. The previous hops' IP router will update their ARP tables (according to RFC826) to send all traffic for that cluster to the new active (previously standby) Network Dispatcher.

Configuring Network Dispatcher

There are many ways that you can configure Network Dispatcher to support your site. If you have only one host name for your site to which all of your customers will connect, you can define a single cluster and any ports to which you want to receive connections. This configuration is shown in Figure 17-1.

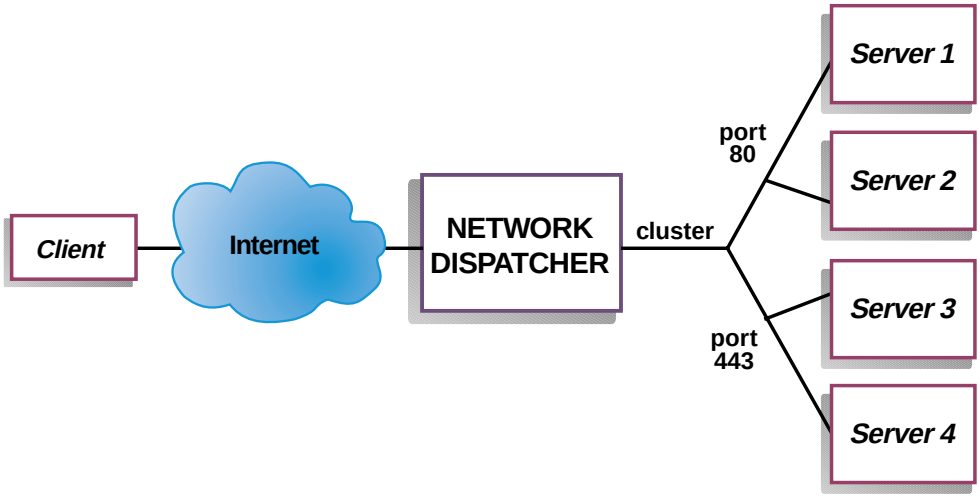


Figure 17-1. Example of Network Dispatcher Configured With a Single Cluster and 2 Ports

Another way of configuring Network Dispatcher would be necessary if your site does content hosting for several companies or departments, each one coming into your site with a different URL. In this case, you might want to define a cluster for each company or department and any ports to which you want to receive connections at that URL as shown in Figure 17-2.

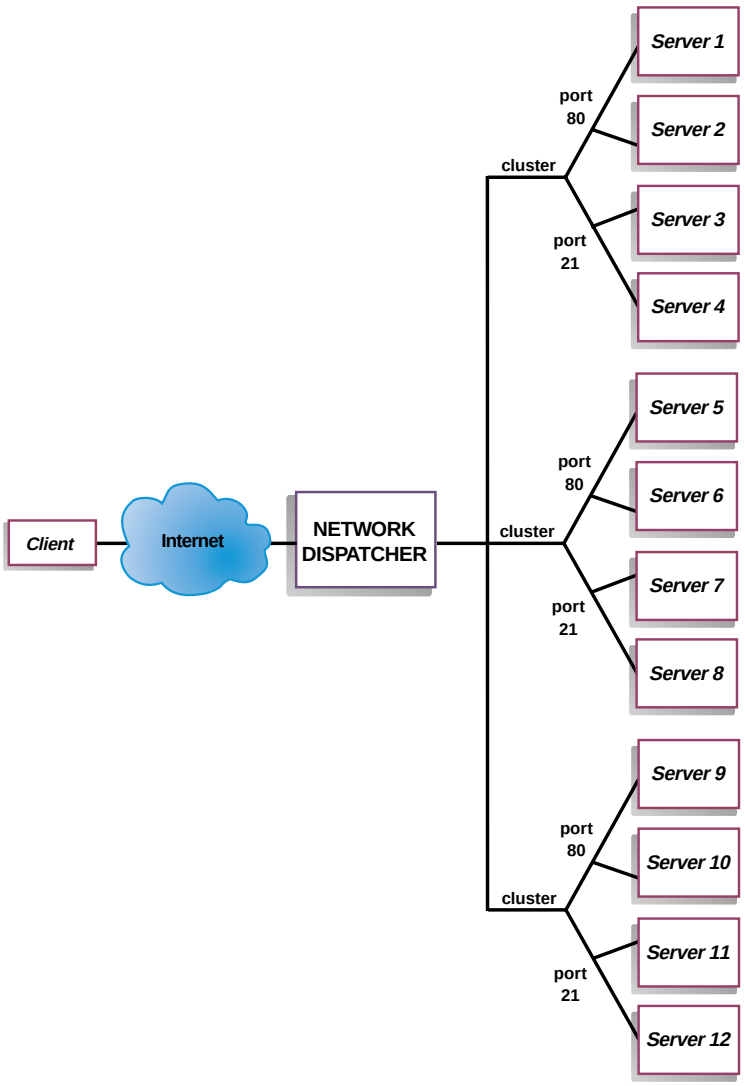


Figure 17-2. Example of Network Dispatcher Configured With 3 Clusters and 3 URLs

A third way of configuring Network Dispatcher would be appropriate if you have a very large site with many servers dedicated to each protocol supported. For example, you may choose to have separate FTP servers with direct T3 lines for large downloadable files. In this case, you might want to define a cluster for each protocol with a single port but many servers as shown in Figure 17-3 on page 17-6.

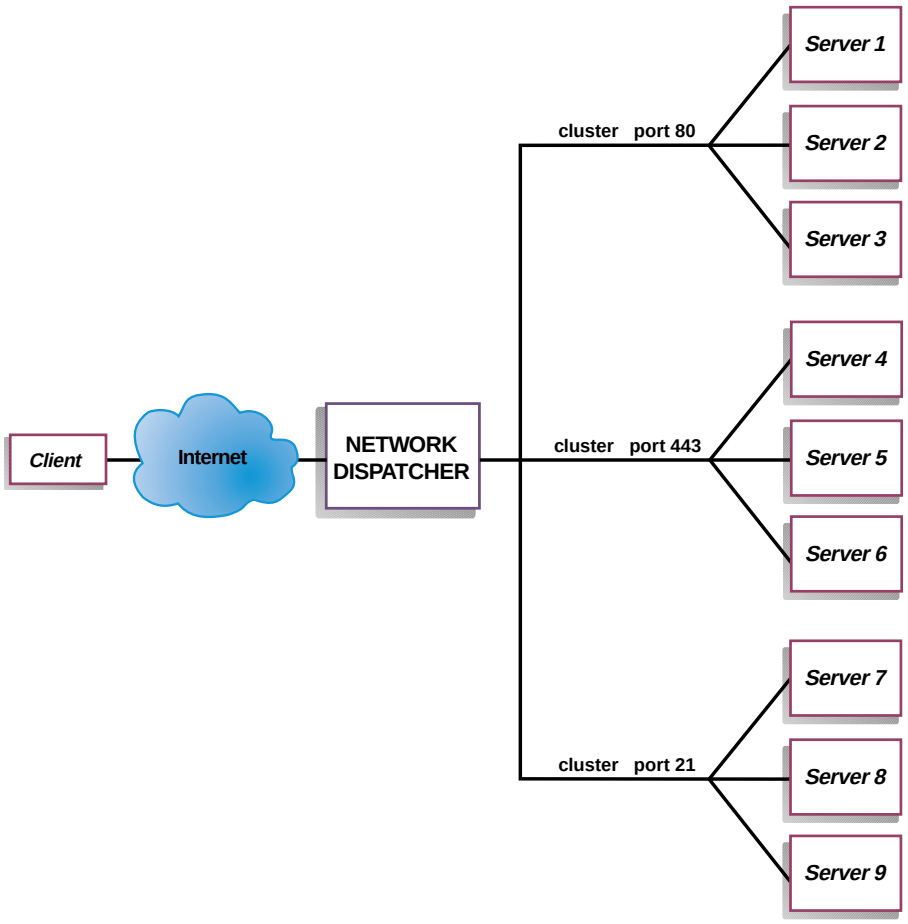


Figure 17-3. Example of Network Dispatcher Configured with 3 Clusters and 3 Ports

Configuration Steps

Before configuring Network Dispatcher:

1. Make sure that the Network Dispatcher has direct interfaces to servers. Servers can have independent connections to the enterprise router or Internet, such that the outgoing traffic from servers to clients can bypass the Network Dispatcher; however, you do not have to configure the independent connection.

   If high availability is important for your network, a typical high availability configuration is shown in Figure 17-4 on page 17-7.

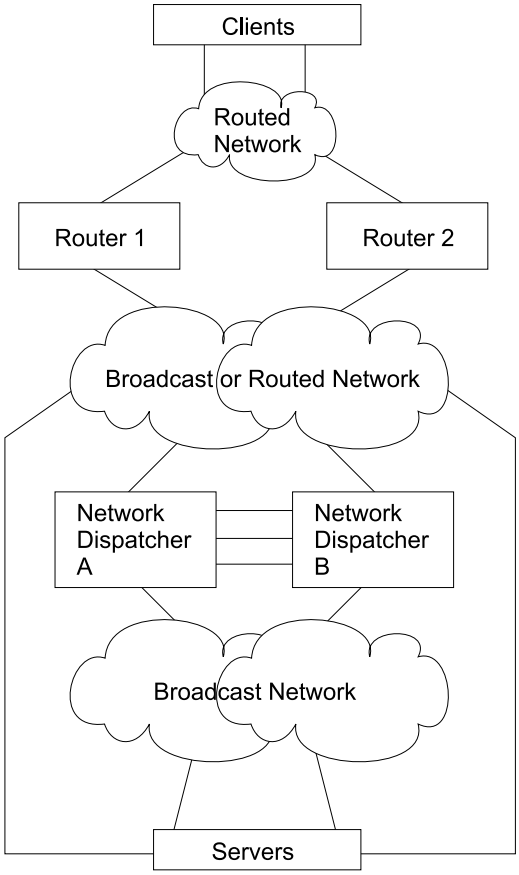


Figure 17-4. High Availability Network Dispatcher Configuration

2. Configure the interfaces of the device. This includes configuring all interfaces, IP addresses on all interfaces, and any applicable routing protocols. You must also configure an internal IP address, using the **set internal-ip-address** command. This is required if you plan to use the Manager and Advisors components. See *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1* for more information about the **set internal-ip-address.**
3. Reboot or restart the device.

Configuring Network Dispatcher on a IBM 2210

To configure Network Dispatcher on a IBM 2210:

1. Access the Network Dispatcher feature, using the **feature ndr** command.
2. Enable the executor and the manager using the **enable executor** and **enable manager** commands.
3. Configure the clusters using the **add cluster** command.
4. Configure the TCP destination ports using the **add port** for each cluster of servers that will serve the corresponding protocol. Examples of the ports are: 80 for HTTP, 20 or 21 for FTP, and 23 for telnet.
5. Configure the servers using the **add server** commands. A server is always associated with a port and a cluster. A server can serve more than one port, a port can be served on more than one server, and a server can belong to more than one cluster, if the server's operating system supports aliasing.

6. Configure any advisors using the **add advisor** command.
7. Enable the advisors that you configured using the **enable advisor** command.

If you are configuring the Network Dispatcher for high availability, continue with the following steps. Otherwise, you have completed the configuration.

Note: Perform these steps on the primary Network Dispatcher and then on the backup.

8. Configure whether this Network Dispatcher is a primary or backup and whether the switchover is manual or automatic using the **add backup** command.
9. Configure all paths (more than one is recommended) on which the heartbeat is going to take place, using the **add heartbeat** command. A path is specified by source and destination IP addresses. The heartbeat is the router that sends the Keepalive messages.
10. Configure the list of host IP addresses that the Network Dispatcher must be able to reach in order to insure a full service, using the **add reach** command. Typically, this will be a subset of servers, the enterprise router, or an administration station.

You can change the configuration using the **set**, **remove**, and **disable** commands.

Configuring Network Dispatcher on a Server

To configure the Network Dispatcher on a server:

1. Alias the loopback device.

   For the TCP servers to work, you must set (or preferably alias) the loopback device (usually called ***lo0***) to the cluster address. Network Dispatcher does not change the destination IP address in the TCP/IP packet before forwarding the packet to a TCP server machine. When you set or alias the loopback device to the cluster address, the TCP server machine will accept a packet that was addressed to another machine.

   If you have an operating system that supports network interface aliasing such as AIX, Solaris, or Windows NT, you should alias ***lo0*** to the cluster address. The benefit of using an operating system that supports aliases is that you can configure the TCP server machines to serve multiple cluster addresses.

   If you have a server with an operating system that does not support aliases, such as HP-UX and OS/2, you must set ***lo0*** to the cluster address.

2. Check for an extra route.

   The network mask for the loopback device is usually 255.0.0.0, so a default route will probably be created. This route needs to be removed.

   Check for an extra route on Windows NT with the **route print** command.

   Check for an extra route on all UNIX systems and OS/2 with the **netstat -nr** command.

3. Delete any extra routes.

   Use the command from Table 17-1 on page 17-9 for your operating system to delete any extra routes.

Table 17-1. Commands to Delete Routes for Various Operating Systems	
Operating System	**Command**
AIX	**route delete -net** *network_address cluster_address*
HP-Unix	**route delete net** *cluster_address*
Solaris	No need to delete route.
OS/2	No need to delete route.
Windows NT	**route delete** *network_address cluster_address* **Note:** This command should be entered at an MS-DOS prompt.

Accessing the Network Dispatcher Configuration Commands

To access the Network Dispatcher configuration environment:

1. Enter **talk 6** at the OPCON prompt (*).
2. Enter **feature ndr** at the GWCON prompt (+).`Config >` prompt.

Network Dispatcher Configuration Commands

Table 17-2 summarizes the Network Dispatcher configuration commands and the rest of the section explains these commands. Enter these commands at the `NDR Config >` prompt.

Table 17-2. Network Dispatcher Configuration Commands	
Command	**Function**
? (Help)	Displays all the Network Dispatcher commands or lists the options for specific commands (if available).
Add	Configures various components of the Network Dispatcher including advisors, clusters, ports, and servers.
Clear	Clears the entire Network Dispatcher configuration.
Disable	Disables the backup, executor, and manager components of the Network Dispatcher. Also disables specific advisors.
Enable	Enables the backup, executor, and manager components of the Network Dispatcher. Also enables specific advisors.
List	Displays the entire Network Dispatcher Configuration or specific portions of the configuration.
Remove	Removes specific portions of the Network Dispatcher configuration.
Set	Changes the configuration parameters for advisors, clusters, ports, servers, or the Network Dispatcher manager.
Exit	Returns you to the previous command level.

? (Help)

Use the **? (Help)** command to display all of the network dispatcher commands or the parameter for a specific command.

Syntax: ?

Example: `NDR Config>` **?**

```
add
clear
disable
enable
list
remove
set
exit
```

Example: `NDR Config>` **`list ?`**

```
all
advisors
backup
cluster
manager
port
servers
```

Add

Use the **add** command to configure advisors, clusters, ports, servers, and to specify which hosts or subnets are reachable through the Network Dispatcher. You can also configure whether this Network Dispatcher is a primary or backup and, if a backup, which server to monitor for recovery.

Syntax: add
- advisor . . .
- backup . . .
- cluster . . .
- heartbeat . . .
- port . . .
- reach . . .
- server . . .

Advisor *name port interval timeout*

Specifies the name and port for an advisor. This parameter also specifies how frequently the advisor will collect information on a particular protocol and a time period after which the advisor considers the protocol unavailable.

name Specifies the type of advisor.

Valid values: 0, 1, 2

0 = FTP

1 = HTTP

2 = MVS

Default value: 1

port Specifies the port number for this advisor.

Valid values: 0 to 65535

Default values:

For advisor 0 – 21

For advisor 1 – 80

For advisor 10007 – 0

interval

Specifies the frequency, in seconds, with which the advisor queries its protocol for each server. After half of this value without a response from the server, the adviser considers the protocol unavailable.

Valid values: 0 to 65535

Default value: 5

timeout

Specifies the interval of time, in seconds, after which the advisor considers the protocol unavailable.

To make sure that out-of-date information is not used by the manager in its load-balancing decisions, the manager will not use information from the advisor whose time stamp is older than the time set in this parameter. The advisor timeout should be larger than the advisor polling interval. If the timeout is smaller, the manager will ignore reports that should be used. By default, advisor reports do not time out.

This timeout value typically applies if you disable an advisor. Do not confuse this parameter with the interval/2 timeout previously described, which relates to a server not responding.

Valid values: 0 to 65535

Default value: 0, which means the protocol is considered always available.

Example: `add advisor`

```
Advisor name (0=ftp, 1=http, 2=mvs) [1]? 1
Port number [80]?
Interval (seconds) [5]? 10
Timeout (0=unlimited) [0]? 10
```

backup *role strategy*

Specifies whether this Network Dispatcher is a backup or primary. If it is a backup, you must also configure a ***heartbeat*** address.

role Defines whether this is a primary or a backup Network Dispatcher. Use this command only if you intend to have a redundant configuration, and want the High Availability function to run. In this case, you must also configure the heartbeat (**add heartbeat**) and reachability (**add reach**).

Valid values: 0 or 1

0 = primary

1 = backup

Default value: 0

strategy

Specifies whether the Network Dispatcher will switch back to primary mode automatically or manually. Whenever a Primary Network Dispatcher fails and become standby (which means a backup performed the IP takeover function), and then becomes available, it will automatically become the active Network Dispatcher if the strategy is set to ***automatic***, as soon as the caches are synchronized. If strategy is set

to ***manual***, the old primary will go to standby mode and the operator must use the **switchover** command to make it active again. See "Switchover" on page 18-8.

Valid values: 0 or 1

0 = automatic

1 = manual

Default value: 0

Example: `add backup`

```
Role (0=Primary, 1=Backup) [Primary]?
Switch back strategy (0=Auto, 1=Manual) [Auto]?
```

cluster *address FIN-count FIN-timeout FIN-stale-timer*

Specifies a cluster's IP address and the frequency for the executor to perform garbage collection from the Network Dispatcher database.

address

Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

FIN-count

Specifies the number of connections that must be in FIN state before the executor tries to remove the unused connection information from the Network Dispatcher database after ***FIN-timeout*** has elapsed.

Valid Values: 0 to 65535

Default value: 4000

FIN-timeout

Specifies the number of seconds, that a connection has been in the FIN state, after which the executor tries to remove the unused connection information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 30

FIN-stale-timer

Specifies the number of seconds, that a connection has been inactive, after which the executor tries to remove a connection's information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 1500

Example: `add cluster`

```
Cluster address [0.0.0.0]? 131.2.24.91
FIN count [4000]?
FIN timeout [30]?
FIN stale timer [1500]?
```

heartbeat *address1 address2*

Specifies one path for Keepalive messages. It is recommended that you configure more than one entry for reliable behavior. The Keepalive message will flow from ***address1***, which belongs to this Network Dispatcher, to ***address2***, which belongs to the peer Network Dispatcher.

address1
Specifies the IP address of the interface of this Network Dispatcher from which Keepalive messages will flow.

Valid Values: Any IP address.

Default value: 0.0.0.0

address2
Specifies the IP address of the interface of the peer Network Dispatcher to which Keepalive messages will flow. This address must be reachable from the interface specified in ***address1.***

Valid Values: Any IP address.

Default value: 0.0.0.0

Example: `add heartbeat`

```
Source Heartbeat address [0.0.0.0]? 131.2.25.90
Target Heartbeat Address [0.0.0.0]? 131.2.25.92
```

port *cluster-address port# max-weight port-mode*
Specifies the cluster and the cluster's attributes.

cluster-address
Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port#
Specifies the port number of the protocol for this cluster.

Valid Values: 0 to 65535

Default value: 80

port-mode
Specifies whether the port will feed all requests from a single client to a single server (known as sticky), use passive ftp (pftp), or use no particular protocols on this cluster (none).

Valid Values: sticky, pftp, or none

Default value: none

max-weight
Specifies the maximum weight for servers on this port. This affects how much difference there can be between the number of requests the executor will give each server.

Valid Values: 0 to 100

Default value: 20

Example: `add port`

```
Cluster address [0.0.0.0]? 131.2.25.91
Port number [80]? 80
Maximum weight (0-100) [20]? 35
Port mode (none=0, sticky=1, pftp=2) [0]?
```

reach *address*

Specifies any host address that the Network Dispatcher must be able to reach to run correctly. It can be a server address, a router address, an administration station address or other IP host.

address

Specifies the target IP address.

Valid Values: Any IP address

Default value: 0.0.0.0

Example: `add reach`

```
Address to reach [0.0.0.0]?
```

server *cluster-address port# server-address server-weight server-state*

Specifies the attributes of a server in a cluster.

cluster-address

Specifies the IP address of the cluster to which this server belongs.

Valid Values: Any IP address

Default value: 0.0.0.0

port#

Specifies the protocol running over the connection to this server.

Valid Values: 0 to 65535

Default value: 80

server-address

Specifies the IP address of the server.

Valid Values: Any IP address

Default value: 0.0.0.0

server-weight

Specifies the weight of the server for the executor. This affects how frequently the Network Dispatcher sends requests to this particular server.

Valid Values: 0 to the value of ***max-weight*** specified on the add port command.

Default value: max-weight on port command

server-state

Specifies whether the executor should regard the server as available or unavailable when the executor begins processing.

Valid Values: 0 (down) or 1 (up)

Default value: 1

Example: `add server`

```
Cluster address [0.0.0.0]? 131.2.25.91
Port number [80]? 80
Server address [0.0.0.0]? 131.2.25.94
Server weight [35]?
Server state (down=0 up=1) [1]?
```

Parameter Configuration Limits

Table 17-3 lists the limits for the various items you can configure for a Network Dispatcher.

Table 17-3. Parameter Configuration Limits

Parameter	Limit
Advisors	8 per physical unit
Clusters	32 per physical unit
Heartbeats	32 per physical unit
Ports	8 per physical unit
Reachs	32 per physical unit
Servers	8 per cluster

Clear

Use the **clear** command to clear the entire Network Dispatcher configuration.

Syntax: clear

Disable

Use the **disable** command to disable a Network Dispatcher component.

Syntax: disable advisor . . .
backup
executor
manager

advisor *name port*

Disables an advisor from the Network Dispatcher.

name Specifies the type of advisor.

Valid values: 0, 1, 2

0 = FTP

1 = HTTP

2 = MVS

Default value: 0

port Specifies the port number for this advisor.

Valid values: 0 to 65535

Default value: 0

Example: `disable advisor`

```
Advisor name (0=ftp, 1=http, 2=mvs) [1]? 1
Port number [0]? 80
```

backup

Disables the Network Dispatcher's backup function.

Example: `disable backup`

```
Backup is now disabled.
```

executor

Disables the Network Dispatcher executor. Disabling the executor disables the Network Dispatcher.

Example: `disable executor`

```
Network dispatcher executor is disabled.
```

Note: Disabling the executor will stop the manager, advisors, and the high availability function, if they are currently running.

manager

Disables the Network Dispatcher manager. The manager is an optional component. However, if you do not use the manager, the Network Dispatcher will balance the load using a round-robin scheduling method based on the current server weights.

Example: `disable manager`

```
Network dispatcher manager is disabled.
```

Note: Because the manager component is prerequisite for advisors, disabling the manager will stop all the advisors from running.

Enable

Use the **enable** command to enable a Network Dispatcher component.

Syntax: enable advisor . . .
backup
executor
manager

advisor *name port*

Enables an advisor to the Network Dispatcher.

name Specifies the type of advisor.

Valid values: 0, 1, 2

0 = FTP

1 = HTTP

2 = MVS

port Specifies the port number for this advisor.

Valid values: 0 to 65535

Default value: 0

Example: `enable advisor`

```
Advisor name (0=ftp, 1=http, 2=mvs) [1]? 1
Port number [0]? 80
```

Note: Because the manager component is a prerequisite for the advisor, you must enable the manager before any advisor can be enabled. You must also set the internal ip address using the **set internal-ip-address** command for the advisor to run correctly. See *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1* for more information about the **set internal-ip-address.** command.

backup

Enables the Network Dispatcher's backup function.

Example: `disable backup`

Note: Before enabling backup, you must add at least one heartbeat

executor

Enables the Network Dispatcher executor.

Example: `enable executor`

```
Network dispatcher executor is enabled.
```

manager

Enables the Network Dispatcher manager.

Example: `enable manager`

```
Network dispatcher manager is enabled.
```

When the manager is enabled for the first time, a manager record is created with the following default values:

Interval: 2 seconds
Refresh-Cycle: 2
Sensitivity: 5 %
Smoothing: 1.5
Proportions:

- **Active:** 50%
- **New:** 50%
- **Advisor:** 0
- **System:** 0

See “Set” on page 17-21 for a description of the above parameters.

List

Use the **list** command to display information about the Network Dispatcher.

Syntax: list
- all
- advisors
- backup
- cluster
- manager
- ports
- servers

all Displays all Network Dispatcher configuration information. This includes the same information displayed for advisors, backup, cluster, manager, ports, and servers.

Example: `NDR Config>` **`list all`**

```
Executor: Enabled

Manager: Enabled

        Interval        Refresh-Cycle   Sensitivity     Smoothing
        2               2               5 %             1.50
        Proportions:    Active  New     Advisor         System
                        50 %    50 %    0 %             0 %

Advisor:
        Name    Port    Interval        TimeOut         State
        http    80      5               0               Enabled
        MVS     10007   15              0               Enabled

Backup: Enabled
        Role            Strategy
        PRIMARY         AUTOMATIC

        Reachability:   Address         Mask            Type
                        131.2.25.93     255.255.255.255 HOST
                        131.2.25.94     255.255.255.255 HOST

        HeartBeat Configuration:
           Source Address:  131.2.25.90 Target Address:   131.2.25.92
           Source Address:  132.2.25.90 Target Address:   132.2.25.92

Clusters:
        Cluster-Addr    FIN-count       FIN-timeout     Stale-timer
        131.2.25.91     4000            30              1500

Ports:
        Cluster-Addr    Port#   Weight  Port-Mode
        131.2.25.91     23      20 %    none
        131.2.25.91     80      20 %    none

Servers:
        Cluster-Addr    Port#   Server-Addr     Weight  State
        131.2.25.91     23      131.2.25.93     20 %    up
        131.2.25.91     23      131.2.25.94     20 %    up
        131.2.25.91     80      131.2.25.93     20 %    up
        131.2.25.91     80      131.2.25.94     20 %    up
```

advisors
Displays the configuration for the Network Dispatcher advisors.

backup
Displays the backup configuration for the Network Dispatcher.

cluster
Displays the configuration of the Network Dispatcher clusters.

manager
Displays the configuration of the Network Dispatcher manager.

ports
Displays the configuration of the Network Dispatcher ports.

servers
Displays the configuration of the servers associated with the Network Dispatcher clusters.

Remove

Use the **remove** command to delete part of the Network Dispatcher configuration.

Syntax: remove advisor . . .
backup
cluster . . .
heartbeat . . .
port . . .
reach . . .
server . . .

advisor *name port*

Removes a specific advisor from the Network Dispatcher configuration.

name Specifies the type of advisor.

Valid values: 0, 1, 2

0 = FTP

1 = HTTP

2 = MVS

port Specifies the port number for this advisor.

Valid values: 0 to 65535

Default value: 0

Example: `remove advisor`

```
Advisor name (0=ftp, 1=http, 2=mvs) [1]?
Advisor port [0]? 80
```

backup

Removes the high availability function.

Note: Because backup is a prerequisite for the heartbeat and reach functions removing backup will stop heartbeat and reach from running.

Example: `remove backup`

cluster *address*

Removes a cluster from the Network Dispatcher configuration.

address Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

Note: Removing a cluster address also removes all the ports and servers associated with that cluster.

Example: `remove cluster`

```
WARNING: Deleting a cluster will make any port or server
         associated with it to also be deleted.
Cluster address [0.0.0.0]? 131.2.25.91
```

heartbeat *address*

Removes the heartbeat server from the Network Dispatcher configuration.

address Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

Example: `remove heartbeat`

```
Target address [0.0.0.0]? 131.2.25.92
```

port *cluster-address port#*

Removes a port from a specific cluster in the Network Dispatcher configuration.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port#

Specifies the port number of the protocol for this cluster.

Valid Values: 0 to 65535

Default value: 0

Note: Removing a port will also remove all of the servers associated with that port.

Example: `remove port`

```
WARNING: Deleting a port will make any server
         associated with it to also be deleted.
Cluster address [0.0.0.0]? 7.82.142.15
Port number [0]? 80
```

reach *address*

Removes a server from the list of hosts the Network Dispatcher must be able to reach.

address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

Example: `remove reach`

```
Target address [0.0.0.0]? 9.82.142.15
```

server *cluster-address port# server-address*

Removes a server from a cluster and port in the Network Dispatcher configuration.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port#

Specifies the port number of the protocol for this cluster.

Valid Values: 0 to 65535

Default value: 80

server-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

Example: `remove server`

```
Cluster address [0.0.0.0]? 7.82.142.15
Port number [0]? 80
Server address [0.0.0.0]? 20.21.22.15
```

Set

Use the **set** command to change the attributes of an existing advisor, cluster, port, or server. You can also define attributes for the Network Dispatcher manager.

Syntax: set
- advisor . . .
- cluster . . .
- manager . . .
- port . . .
- server . . .

advisor ***name port# interval timeout***

Changes the port number, interval, and timeout for an advisor.

name Specifies the type of advisor.

0 = FTP

1 = HTTP

2 = MVS

Valid values: 0, 1, 2

Default value: 1

port Specifies the port number for this advisor.

Valid values: 0 to 65535

Default value: 0

interval

Specifies the frequency with which the advisor queries its protocol for each server. After half of this value expires without a response from the server, the adviser considers the protocol unavailable.

Valid values: 0 to 65535

Default value: 5

timeout

Specifies the interval of time, in seconds, after which the advisor considers the protocol unavailable.

To make sure that out-of-date information is not used by the manager in its load-balancing decisions, the manager will not use information from the advisor whose time stamp is older than the time set in this parameter. The advisor timeout should be larger than the advisor polling interval. If the timeout is smaller, the manager will ignore reports that should be used. By default, advisor reports do not time out.

This timeout value typically applies if you disable an advisor. Do not confuse this parameter with the interval/2 timeout previously described, which relates to a server not responding.

Valid values: 0 to 65535

Default value: 0, which means the protocol is considered always available.

Example: `set advisor`

```
Advisor name (0=ftp, 1=http, 2=mvs) [0]?
Port for advisor [0]? 23
Interval (seconds) [5]? 10
Timeout (0=unlimited) [0]? 20
```

cluster *address FIN-count FIN-timeout FIN-stale-timer*

Changes the FIN-count, FIN-timeout, and FIN-stale-timer for a cluster in the Network Dispatcher configuration.

address

Specifies the IP address for the cluster.

Valid values: Any valid IP address

Default value: 0.0.0.0

FIN-count

Specifies the number of connections that must be in FIN state before the executor tries to remove the unused connection information from the Network Dispatcher database after ***FIN-timeout*** has elapsed.

Valid Values: 0 to 65535

Default value: 4000

FIN-timeout

Specifies the number of seconds after which the executor tries to remove the unused connection information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 30

FIN-stale-timer

Specifies the number of seconds that a connection has been inactive, after which the executor tries to remove a connection's information from the Network Dispatcher database.

Valid Values: 0 to 65535

Default value: 1500

Example: `set cluster`

```
Cluster address [0.0.0.0]? 131.2.25.91
FIN count [4000]? 4500
FIN timeout [30]? 40
FIN stale timer [1500]? 2000
```

manager *interval proportion refresh sensitivity smoothing*

Sets the values that the manager uses to determine the best server to satisfy a request.

interval

Specifies the amount of time, in seconds, after which the manager updates the server weights that the executor uses in routing connections.

Valid values: 0 to 65535

Default value: 2

proportion
Specifies the relative importance of external factors in the manager's weighting decisions. The sum of the proportions must equal 100%. The factors are:

active
The number of active connections on each TCP/IP server as tracked by the executor.

Valid values: 0 to 100

Default value: 50

new The number of new connections on each TCP/IP server as tracked by the executor.

Valid values: 0 to 100

Default value: 50

advisor
Input from the advisors defined to the Network Dispatcher.

Valid values: 0 to 100

Default value: 0

system
Input from the system monitoring tools.

Valid values: 0 to 100

Default value: 0

refresh
Specifies the frequency with which the manager requests status from the executor. This parameter is specified as a number of ***intervals***.

Valid values: 0 to 100

Default value: 2

sensitivity
Specifies the percentage weight change for all the servers on a port, after which the manager updates the weights that the executor uses in routing connections.

Valid values: 0 to 100

Default value: 5

smoothing
Specifies a limit to the amount that a server's weight can change. Smoothing minimizes the frequency of change in the distribution of requests. A higher smoothing index will cause the weights to change less. A lower smoothing index will cause the weights to change more.

Valid values: a decimal value between 1.0 and 42 949 673.00

Default value: 1.5

Note: You can only specify two places after the decimal point.

Example: `set manager`

```
Manager interval (in seconds) [2]? 3
Active proportion [50]? 25
New proportion [50]? 25
Advisor proportion [0]? 25
System proportion [0]? 25
Refresh cycle [2]? 4
Sensitivity threshold [5]? 10
Smoothing index (>1.00) [1.50]? 200
```

port *cluster-address port# weight*

Changes the port number and weight for a specific cluster.

cluster-address

Specifies the IP address of the cluster.

Valid Values: Any IP address.

Default value: 0.0.0.0

port#

Specifies the port number of the protocol for this cluster.

Valid Values: 0 to 65535

Default value: 80

weight

Specifies the weight for servers on this port. This affects how much difference there can be between the number of requests the executor will give each server.

Valid Values: 1 to 100

Default value: 20

Example: `set port`

```
Cluster address [0.0.0.0]? 131.2.25.91
Port number [0]? 23
Max. weight (0-100) [20]? 30
```

server *cluster-address port# server-address weight state*

Changes the port number, server address, server state, and server weight for a specific server in a cluster.

cluster-address

Specifies the IP address of the cluster to which this server belongs.

Valid Values: Any IP address

Default value: 0.0.0.0

port#

Specifies the protocol running over the connection to this server.

Valid Values: 0 to 65535

Default value: 80

server-address

Specifies the IP address of the server.

Valid Values: Any IP address

Default value: 0.0.0.0

state

Specifies whether the executor should regard the server as available or unavailable when the executor begins processing.

Valid Values: 0 (down) or 1 (up)

Default value: 1

weight

Specifies the weight of the server for the executor. This affects how frequently the Network Dispatcher sends requests to this particular server.

Valid Values: 0 to the value of ***max-weight*** specified on the add port command.

Default value: max-weight on port command

Example: `set server`

```
Cluster address [0.0.0.0]? 131.2.25.91
Port number [0]? 80
Server address [0.0.0.0]? 131.2.25.94
Server weight [20]? 25
Server state (down=0, up=1) [1]? 1
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **exit**

Chapter 18. Monitoring the Network Dispatcher Feature

This chapter describes how to monitor the Network Dispatcher Feature. It contains the following sections:

- "Accessing the Network Dispatcher Monitoring Commands"
- "Network Dispatcher Monitoring Commands"

Accessing the Network Dispatcher Monitoring Commands

To access the Network Dispatcher monitoring environment:

1. Enter **talk 5** at the OPCON prompt (*).
2. Enter **feature ndr** at the GWCON prompt (+).

Network Dispatcher Monitoring Commands

Table 18-1 summarizes the Network Dispatcher monitoring commands and the rest of the section explains these commands. Enter these commands at the `NDR >` prompt.

Table 18-1. Network Dispatcher Monitoring Commands

Command	Function
? (Help)	Displays all the Network Dispatcher commands or lists the options for specific commands (if available).
List	Displays the currently configured attributes of the advisor, clusters, ports, or servers.
Quiesce	Specifies that no more connection request should be sent to a server. Also temporarily stops the heartbeat and reach functions.
Report	Displays a report of information related to the advisor and the manager.
Status	Displays the current status of the counters, clusters, ports, and servers, advisor, manager, and backup.
Switchover	Forces a Network Dispatcher that is running in standby mode to become the active Network Dispatcher. Use this command is necessary f you specified manual as the switchover mode.
Unquiesce	Allows the Network Dispatcher manager to assign a weight greater than 0 to a previously quiesced server on every port that the server is configured. This action allows new connection requests to flow to the selected server.
Exit	Returns you to the previous command level.

? (Help)

Use the **? (Help)** command to display all of the network dispatcher commands or the parameter for a specific command.

Syntax: ?

Example: `NDR >` **?**

```
list
quiesce
report
status
switchover
trace
unquiesce
exit
```

Example: NDR > **list ?**

```
advisor
cluster
port
server
```

List

Use the **list** command to display information about the Network Dispatcher.

Syntax: list advisor
cluster
ports
servers

advisor

Displays the configuration for the Network Dispatcher advisors.

Example: list advisor

```
Advisor list requested.

-----------------------------------------
| ADVISOR |  PORT  |  TIMEOUT  |  STATUS  |
-----------------------------------------
|  ftp    |    23  |       5   |   ACTIVE |
|  Http   |    80  | unlimited |   ACTIVE |
|  MVS    | 10007  | unlimited |   ACTIVE |
-----------------------------------------
```

cluster

Displays the configuration of the Network Dispatcher clusters.

Example: list cluster

```
EXECUTOR INFORMATION:
 --------------------
Version: 01.01.00.00 - Tue Dec 10 14:15:58 EST 1996
Number of defined clusters: 2

CLUSTER LIST:
-------------
  131.2.25.91
  10.11.12.2
```

ports

Displays the configuration of the Network Dispatcher ports.

Example: list ports

```
Cluster Address [0.0.0.0]? 131.2.25.91

-------------------------------------------
| CLUSTER:       131.2.25.91              |
|-----------------------------------------|
|   PORT   |    MAXWEIGHT   |  STICKY/PFTP  |
|-----------------------------------------|
|     23   |         30     |    neither    |
|     80   |         20     |    neither    |
-------------------------------------------
```

server
Displays the configuration of the servers associated with the Network Dispatcher clusters.

Example: `list servers`

```
Cluster Address [0.0.0.0]? 131.2.25.91

PORT 23 INFORMATION:
 --------------------
Maximum weight.................. 20
Port is sticky.................. FALSE
Port is for passive ftp......... FALSE
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1

PORT 80 INFORMATION:
 --------------------
Maximum weight.................. 20
Port is sticky.................. FALSE
Port is for passive ftp......... FALSE
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1
```

Quiesce

Use the **quiesce** command to temporarily stop the heartbeat or reach functions or to specify that no more connection requests should be sent to a server.

Syntax: quiesce heartbeat
manager
reach

heartbeat *address*
Stops the selected path for the heartbeat function. The ***address*** is the IP address of the remote server to which this Network Dispatcher is sending Keepalive messages.

Example: `quiesce heartbeat`

```
Remote Address [0.0.0.0]? 131.2.25.94
```

manager *address*
Specifies that no more connection requests are to be made to the specified server. ***Address*** is the IP address of the server.

Example: `quiesce manager`

```
Server Address [0.0.0.0]? 131.2.25.93
```

reach *address*
Stops the Network Dispatcher's polling of the specified address to determine if it is reachable, where ***address*** is the IP address that is part of the reachability criteria.

Example: `quiesce reach`

```
Reach Address [0.0.0.0]? 131.2.25.92
```

Report

Use the **report** command to display a report of the advisor or manager

Syntax: report advisor
manager

advisor *type port#*

Displays a report of information about a specific advisor.

type Is the type of advisor: 0 = ftp, 1 = http, 2 = MVS.

port# Is the port number.

Example: `report advisor`

```
0=ftp, 1=http, 2=MVS
Advisor name [0]? 1
Port number [0]? 80

------------------------------
|      ADVISOR:     http     |
|      PORT:          80     |
------------------------------
| 131.2.25.93      |       0  |
| 131.2.25.94      |      16  |
------------------------------
```

manager

Displays a report of the current manager information.

Example: `report manager`

```
---------------------------------
|  HOST TABLE LIST  |   STATUS   |
---------------------------------
|  131.2.25.93      |    ACTIVE  |
|  131.2.25.94      |    ACTIVE  |
---------------------------------

-------------------------------------------------------------------------------
|131.2.25.91    |WEIGHT | ACTIVE % 50  |  NEW % 50    | PORT %  0 |SYSTEM %  0|
|               ---------------------------------------------------------------|
|PORT:   23     |NOW|NEW| WT | CONNECT | WT | CONNECT | WT | LOAD | WT | LOAD |
|-------------------------------------------------------------------------------|
|131.2.25.93    | 10| 10|  10|        0|  10|        0|   0|     0|-999|    -1|
|131.2.25.94    | 10| 10|  10|        0|  10|        0|   0|     0|-999|    -1|
|-------------------------------------------------------------------------------|
|PORT TOTALS:   | 20| 20|    |        0|    |        0|    |     0|    |    -2|
-------------------------------------------------------------------------------

-------------------------------------------------------------------------------
|131.2.25.91    |WEIGHT | ACTIVE % 50  |  NEW % 50    | PORT %  0 |SYSTEM %  0|
|               ---------------------------------------------------------------|
|PORT:   80     |NOW|NEW| WT | CONNECT | WT | CONNECT | WT | LOAD | WT | LOAD |
|-------------------------------------------------------------------------------|
|131.2.25.93    | 10| 10|  10|        0|  10|        1|  16|     0|-999|    -1|
|131.2.25.94    | 10| 10|  10|        0|  10|        1|   3|    16|-999|    -1|
|-------------------------------------------------------------------------------|
|PORT TOTALS:   | 20| 20|    |        0|    |        0|    |    16|    |    -2|
-------------------------------------------------------------------------------

-----------------------------------------
| ADVISOR |  PORT  |  TIMEOUT  |  STATUS  |
-----------------------------------------
|  http   |    80  | unlimited |   ACTIVE |
|  MVS    | 10007  | unlimited |   ACTIVE |
-----------------------------------------
Manager report requested.
```

Status

Use the **status** command to obtain the status of the advisors, backup, counter, clusters, manager, ports, and servers.

Syntax: status
- advisor
- backup
- cluster
- counter
- manager
- port
- server

advisor *type port#*

Obtains the status of a specific advisor.

type Is the type of advisor. 0 = ftp, 1 = http, 2 = MVS.

port# Is the port number.

Example: `status advisor`

```
0=ftp, 1=http, 2=MVS
Advisor name [0]?
Port number [0]? 23

Advisor ftp on port 23 status:
==============================
Logging level..... 0
Interval.......... 10
```

backup

Obtains the status of the backup function.

Example: `status backup`

```
 Dumping status ...
 Role : PRIMARY   Strategy : AUTOMATIC  State : ND_ACTIVE Sub-State : ND_NOT_SYNCHRONIZED
 <<Preferred Target : 132.2.25.92>>

 Dumping HeartBeat Status ...
 .....Heartbeat target : 131.2.25.92 Status : UNREACHABLE
 .....Heartbeat target : 132.2.25.92 Status : REACHABLE

 Dumping Reachability Status ...
 .....Host:131.2.25.93   Local:REACHABLE
 .....Host:131.2.25.94   Local:REACHABLE
```

cluster *address*

Obtains the status of a specified cluster, where ***address*** is the IP address of the cluster.

Example: `status cluster`

```
Cluster Address [0.0.0.0]? 131.2.25.91

EXECUTOR INFORMATION:
 --------------------
Version: 01.01.00.00 - Tue Dec 10 14:15:58 EST 1996

CLUSTER INFORMATION:
 --------------------
Address........................ 131.2.25.91
Number of target ports.......... 2
FIN clean up count.............. 4000
Connection FIN timeout.......... 30
Active connection stale timer... 1500

PORT 23 INFORMATION:
 --------------------
Maximum weight.................. 20
Port is sticky.................. FALSE
Port is for passive ftp......... FALSE
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1

PORT 80 INFORMATION:
 --------------------
Maximum weight.................. 20
Port is sticky.................. FALSE
Port is for passive ftp......... FALSE
All up nodes are weight zero.... FALSE
Total target nodes.............. 2
Currently marked down........... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 0 Active: 0 FIN 0 Status: up Saved Weight: -1
```

counter

Obtains the status of all counters.

Example: `status counter`

```
Internal counters from executor:
--------------------------------
Total number of packets into executor...... 2684
Discarded because headers too short........ 0
Packets to non forwarding address.......... 0
Total packets for cluster processing (C)... 2684
Packets not addressed to a cluster(port)... 0

Cluster processing results:
---------------------------
Errors..................................... 0
Discarded.................................. 0
Own address................................ 0
Forward requested.......................... 2684
Forward discarded with error............... 0

Other processing problems:
--------------------------
Total packets dropped (C).................. 0
```

manager

Obtains the status of the manager.

Example: `status manager`

```
Number of defined hosts... 2
Logging level..... 0
Sensitivity..... 0%
Smoothing factor..... 2
Interval..... 3
Weights refresh cycle..... 4

Active connections gauge proportion..... 50%
New connections counter(delta) proportion... 50%
Advisor gauge proportion.....  0%
System Metric proportion.....  0%

Manager status requested.
```

port *clusteraddress port#*

Obtains the status of a specific port, where:

clusteraddress is the IP address of the port.

port# is the port number on the cluster.

Example: `status port`

```
Cluster Address [0.0.0.0]? 131.2.25.91
Port number  [0]? 80

PORT 80 INFORMATION:
 --------------------
Maximum weight..... 20
Port is sticky..... FALSE
Port is for passive ftp..... FALSE
All up nodes are weight zero.... FALSE
Total target nodes..... 2
Currently marked down..... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 12345 Active: 3431 FIN 3780 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 7890  Active: 2980 FIN 2390 Status: up Saved Weight: -1
```

server *address*

Obtains the status of a specific server, where ***address*** is the IP address of the server.

Example: `status server`

```
Cluster Address [0.0.0.0]? 131.2.25.91

PORT 23 INFORMATION:
 --------------------
Maximum weight..... 20
Port is sticky..... FALSE
Port is for passive ftp..... FALSE
All up nodes are weight zero.... FALSE
Total target nodes..... 2
Currently marked down..... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 140 Active: 50 FIN 45 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 250 Active: 60 FIN 54 Status: up Saved Weight: -1

PORT 80 INFORMATION:
 --------------------
Maximum weight..... 20
Port is sticky..... FALSE
Port is for passive ftp..... FALSE
All up nodes are weight zero.... FALSE
Total target nodes..... 2
Currently marked down..... 0
Servers providing service to this port:
Address: 131.2.25.93 Weight: 20 Count: 12345 Active: 3431 FIN 3780 Status: up Saved Weight: -1
Address: 131.2.25.94 Weight: 20 Count: 7890  Active: 2980 FIN 2390 Status: up Saved Weight: -1
```

Switchover

Use the **switchover** command to force a Network Dispatcher that is running in standby mode to become the active Network Dispatcher when the switchover strategy is manual. This command must be entered on the host that is running the Network Dispatcher that is in standby mode.

Syntax: switchover

Unquiesce

Use the **unquiesce** command to restart a heartbeat, manager, or reach function that was previously stopped with the **quiesce** command.

Syntax: unquiesce heartbeat
manager
reach

heartbeat *address*

Restarts the path for Keepalive messages, where ***address*** is the IP address of the remote server to which this Network Dispatcher is sending Keepalive messages.

Example: `unquiesce heartbeat`

```
Remote Address [0.0.0.0]? 9.10.11.1
```

manager *address*

Restarts sending connection requests to the specified server. ***Address*** is the IP address of the server.

Example: `unquiesce manager`

```
Server Address [0.0.0.0]? 20.21.22.15
```

reach *address*

Restarts the Network Dispatcher's polling of the specified address to determine if it is reachable, where ***address*** is the IP address that is part of the reachability criteria.

Example: `unquiesce reach`

```
Reach address [0.0.0.0]? 20.3.4.5
```

Exit

Use the **exit** command to return to the previous prompt level.

Chapter 19. The Data Compression Subsystem

This chapter discusses data compression on a 2210 over Frame Relay and PPP interfaces. It includes these sections:

Data compression is supported on Frame Relay and PPP interfaces.

Data Compression Overview

The data compression system provides a means to increase the effective bandwidth of networking interfaces on the device. It is primarily intended for use on slower speed WAN links.

Data compression on the device is supported on PPP and Frame Relay interfaces:

- For PPP interfaces, compression is implemented according to the Compression Control Protocol (CCP) as defined in the Internet Engineering Task Force's RFC 1962. CCP provides the underlying mechanisms by which the use of compression is negotiated and a means for choosing among multiple possible compression algorithms or protocols.

  The device provides two compression protocols: the Stac-LZS protocol, defined in RFC 1974; and the Microsoft Point-to-Point Compression protocol (MPPC), described in RFC 2118. Both of these are based on compression algorithms provided by Stac Electronics.

- For Frame Relay interfaces, compression is implemented according to FRF.9, the *Data Compression over Frame Relay Implementation Agreement* produced by the Frame Relay Forum Technical Committee. FRF.9 describes a Data Compression Protocol (DCP), modeled after PPP's CCP, and similarly provides a means for negotiating various compression algorithms and options. The device supports DCP "mode 1" negotiation. FRF.9 also describes a more generalized "mode 2"; this is not supported. Compression itself is done using the same compression engine as used for the PPP Stac-LZS protocol.

Data Compression Concepts

Data compression on the device provides a means to increase throughput on network links by making more efficient use of the available bandwidth on a link. The basic principle behind this is simple: represent the data flowing across a link in as compact a manner as possible so that the time needed to transmit it is as low as possible, given a set speed on a link.

Data compression may be performed at many layers in the networking model. At one end of the spectrum, applications may compress data prior to transmitting it to peer applications elsewhere in the network, while at the other end of the spectrum devices may be performing compression at the data link layer, working purely on the bit stream passing between two nodes. How this compression is done and how effective it is depends on a variety of factors, including such things as what network layer the compression is performed at, how much intrinsic knowledge the

compressor and decompressor have about the data being compressed, the compression algorithm chosen, and the actual data being compressed. The best compression can usually be performed at the application layer; for example, a file transfer application usually has the luxury of having an entire file of data available to it prior to attempting compression, and it may be able to try different compression algorithms on the file to see which performs best on that particular file's data. Although this may provide excellent compression for that one type of application, it does little to solve the general problem of compressing the bulk of the traffic flowing over a network, as most networking applications do not currently compress data as they generate it.

Compression on the device takes place at a much lower networking layer, at the data link layer. In the device, compression is performed on the individual packets which are transmitted across a link. The compression is done in real-time as packets flow through the device: the sender compresses a packet just prior to transmitting it, and the decompressor decompresses the packet as soon as it receives it. This operation is transparent to the higher layer networking protocols.

Data Compression Basics

Data compressors work by recognizing "redundant" information in data, and producing a different set of data which contains as little redundancy as possible. "Redundant" information is any information which can be derived and recreated based on the currently available data. For example, a compressor might function by recognizing repeated character patterns in a data stream and replacing these repeated patterns with a shorter code sequence to represent that pattern. As long as the compressor and decompressor agree on what these code sequences are then the decompressor can always recreate the original data from the compressed data.

This mapping of sequences in the original data to corresponding sequences in the compressed output is commonly called a ***data dictionary***. These dictionaries may be statically defined - experienced-based information available to the compressor and decompressor - or they may be dynamically generated, usually based on the information being compressed. Static dictionaries are most applicable to environments where the data being processed is of a limited, known nature, and not very effective for general-purpose compressors. Most compression systems use dynamic dictionaries, including any compressors used on the device. On a 2210 the data dictionaries are based on the current packet being processed and possibly previously seen packets, but there is no ability to "look ahead" in the data stream as may exist when compression is performed at other layers. For systems where the data dictionary is dynamically derived and based only on previously seen data, the dictionary is also commonly known as a ***history***. The terms history and data dictionary will be used interchangeably throughout the remainder of this chapter, though it should be understood that in other environments a history is a specific form of data dictionary.

The fact that the device uses dynamic dictionaries and that the compressor and decompressor must keep their dictionaries in synchronization means that data compression works on a stream of data passing between two endpoints. Hence, compression on the router is a connection-oriented process, where the endpoints of the connection are the compressor and decompressor themselves. When compression is started on the stream, both ends reset their data dictionaries to some known starting state, and then they update that state as data is received.

Compression could be performed on each individual packet, resetting the histories prior to processing each packet. Normally though, the data dictionaries are not reset between packets, which means that the histories are based not only on the contents of the current packet, but also the contents of previously seen packets. This usually improves the overall compression effectiveness, because it increases the amount of data which the compressor searches looking for redundancy to remove. As an example, consider the case of one host “ping”ing another host with IP: a series of packets is sent out, each one usually nearly identical to the last one sent. The compressor may have little luck compressing the first packet, but it may recognize that each subsequent packet looks very much like the last one sent, and produce highly compressed versions of those packets.

Because the compressor and decompressor histories change with each packet received, the compression mechanisms are sensitive to lost, corrupted, or reordered packets. The compression protocols employed by the device include signalling mechanisms whereby the compressor and decompressor can detect loss of synchronization and resynchronize to each other, such as might be necessary when a packet is lost due to a transmission error. Typically this is done by including a sequence number in each packet which the decompressor will check to make sure it is receiving all packets, in order. If it detects an error, it will reset itself to some known starting state, signal the compressor to do likewise, and then wait (discarding incoming compressed packets) until the compressor acknowledges that it has also reset itself.

Compression on a link typically is performed on data going in both directions over the link. Normally, each end of a connection has both a compressor and decompressor running on it, communicating with their analogs at the other end of the connection, as shown in Figure 19-1 on page 19-4. The output (compression) side runs independently of the input (decompression) side. It is possible for completely different compression algorithms to be operating for each direction of the link. When a link connection is established, the compression control protocol for the link will negotiate with the peer to determine the compression algorithm(s) used for the connection. If the two ends cannot agree on compression protocols to use, then no compression will be performed and the link will operate normally - packets will simply be sent in uncompressed form.

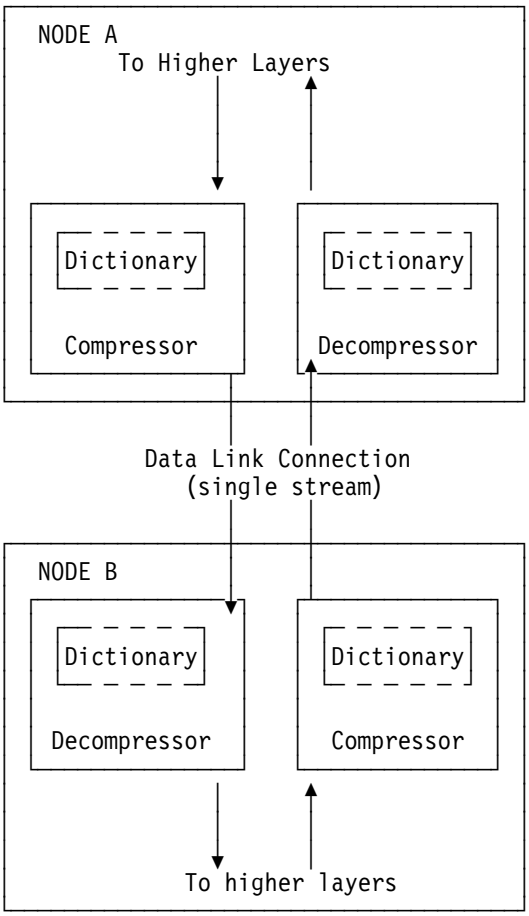


Figure 19-1. Example of Bidirectional Data Compression with Data Dictionaries

A stream really represents a connection between a specific compression process on one end of a link and an associated decompression process on the other end of a link, and thus is more specific than just a “connection” between two nodes; it is possible that a sophisticated compression protocol could split the data flowing between two hosts into multiple streams, compressing each of these streams independently. For example, PPP’s CCP has the ability to negotiate the use of multiple histories over a single PPP link, though the router does not support this.

Considerations

The choice of whether or not to use data compression is not always an easy one. There are several factors which should be considered before enabling compression on a connection.

CPU Load

Data compression is a computationally expensive procedure. As the amount of data being compressed increases (per unit time), the more of a load is put on the device’s processor. If the load becomes too great, the performance of the device degrades - on all network interfaces, not just the ones where compression is being performed.

The device actually contains multiple processors and uses asymmetric multiprocessing - for example, link I/O controllers which operate in tandem with the main processor - so the effect of the processor loading is not always readily

measured. Because the compression operation may be overlapped with the transmission of packets, this loading may in fact be totally transparent and pose no problem. Nonetheless, it is possible to overburden the device's processor and degrade performance.

As a general rule of thumb, compression should only be enabled on slow speed WAN links - probably only for links with speeds up to about 64 kilobits per second (the speed of a typical ISDN dial link). The total bandwidth for data being compressed on all links probably should be limited to several hundred kilobits per second. Running compression on all channels of an ISDN Primary Rate adaptor would be unwise.

Some of the device configuration parameters allow you to limit the number of connections which may be concurrently running compression. More interfaces can be enabled for compression then than are actually running it. Once the limit on the number of active compression connections is reached, additional connections will simply not negotiate the use of compression, at least not until an existing compression link shuts down.

Memory Usage

Another issue to consider when configuring compression is the memory requirement. Compression and decompression histories occupy a fair amount of memory, which is a limited resource in the device. The Stac-LZS algorithm for example requires about 13 Kbytes for a compression history, and about 5 Kbytes for a decompression history. This problem is magnified by the fact that these histories must exist for each connection which is established: a compression history is synchronized with a corresponding decompression history in a peer router. For a PPP link, this implies one compression history and one decompression history (assuming that data compression is running bidirectionally on the link). On a Frame Relay link, there could be many such histories required, one pair for each virtual connection (DLCI) which is established.

The device allocates a limited number of compression and decompression histories when it boots. These are always allocated in pairs known as ***compression contexts*** - a context is simply one compression history coupled with one decompression history. Technically, compression and decompression are independent functions and the allocation of compression and decompression histories could be performed independently; however, in practice compression is almost always run bidirectionally and so memory is managed and configured in terms of contexts rather than individual histories as a way of simplifying operation.

Whenever the device attempts to establish a compression connection on a link, it begins by reserving a context from the allocated pool of contexts. If no contexts are available, then compression is not performed on that connection. The router may attempt to start compression on that connection later as contexts become available.

The number of compression contexts which are allocated is a configurable parameter. Setting the number of contexts allocated limits both the amount of memory used and the maximum number of connections which may be simultaneously operating with compression. Limiting the number of simultaneously operating compression connections provides a means to help control the CPU loading problem.

Data Content

The actual nature of the data being transmitted on a connection should be considered before enabling compression for that connection. Compression works better on some types of data than others. Packets which contain a lot of nearly identical information - for example a set of packets generated from an IP “ping” - will normally compress extremely well. A typical assortment of random text and binary data going over a link will usually compress in ratios around 1.5:1 to 3:1. Some data simply will not compress well at all. In particular, data which has already been compressed will seldom compress further. In fact, data which has been previously compressed may actually expand when fed through the compression engine.

If it is known in advance that most of the data flowing over a connection will consist of compressed data, then it is recommended that compression not be enabled for that connection. An example where this might occur is a connection to a host which was set up to be primarily a FTP file archive site, where all the files available to be transferred are stored in compressed form on the host.

Link Layer Compression

A final factor to consider is the nature of the network link between the two hosts. Compression could be performed at a lower layer than even the device’s hardware interfaces. In particular, many modern modems incorporate data compression mechanisms in their hardware and firmware. If compression is being performed on the link at a lower layer (outside the device), then it is best not to enable data compression on the device for that interface. As already mentioned, compressing an already compressed data stream is normally ineffective, and in fact may degrade performance slightly. Unless there is some particular reason to believe that the router will do a much better job of compression than the link hardware, it is best to let the link hardware do the compression.

Configuring and Monitoring Data Compression

Configuring data compression on a 2210 is a two-step process. The core compression system is a “Feature” in the software. You set and monitor global parameters by selecting the CMPRS feature in the Configuration and Monitoring tasks (the GWCON and CONFIG processes in the router). In addition to configuring the global parameters, you must also configure compression for each network interface (PPP or Frame Relay) on which you will transmit compressed data traffic.

This section describes configuring and monitoring the compression feature first and then describes configuring and monitoring compression on PPP and Frame Relay interfaces.

Configuring the Compression Feature

The only configurable parameter for the compression feature is the number of compression contexts to allocate when the device boots. The number of available contexts limits the number of connections that can be active simultaneously, as well as determining the amount of memory set aside for compression histories. Setting the number of contexts to zero disables compression on all interfaces.

In the Config process, enter **feature cmprs** at the `Config >` prompt to access the compression configuration commands. To change the number of contexts allocated, use the **SET MAXCONTEXTS *n*** command where ***n*** is the number of contexts. To

see the current configuration, use the **list** command. The complete set of configuration commands is summarized in Table 19-1 on page 19-7, and a configuration example is shown in Figure 19-2 on page 19-7.

```
Config> feature cmprs

Data Compression Global Configuration
CMPRS Config> ?
LIST
SET
EXIT

CMPRS Config> set ?
MAXCONTEXTS

CMPRS Config> set maxcontexts
Number of compression contexts to allocate? [0]? 10

CMPRS Config> list
Number of compression contexts to allocate: 10
```

Figure 19-2. Configuring the Compression Feature

Table 19-1. Compression Configuration Commands

Command	Action
? (Help)	Displays the commands available at this command level.
List	Displays the current setting of maxcontexts.
Set	Sets the maximum number of compression contexts available for all interfaces.
Exit	Returns you to the previous prompt level.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
list
set
exit
```

List

Use the **list** command to display the current setting of *maxcontexts*.

Syntax: list

Example: **list**

```
Number of compression contexts to allocate: 10
```

Set

Use the **set** command to set the maximum number of interfaces that can use data compression simultaneously.

Syntax: set maxcontexts *n*

maxcontexts *n* Sets the maximum number of compression contexts available for the interfaces. This parameter causes the device to allocate a pool of memory for compression contexts. Setting maxcontexts to 0 prevents any interface from compressing data even if you enabled compression on the interface.

Note: Setting this value too high can result in excessive memory use and decreased throughput for the device.

Default Value: 0

Valid Values: 0 to 65535

Example: `set maxcontexts`

```
Number of compression contexts to allocate? [0]? 10
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **exit**

Monitoring the Compression Feature

In the Console process, enter **feature cmprs** at the + prompt to access the compression monitoring commands. Table 19-2 lists the available commands.

Table 19-2. Compression Monitoring Command

Command	Action
? (Help)	Displays the commands available at this level.
List	List either the memory or contexts in use.
Exit	Return to the previous command level.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
list
```

List

Use the **list** command to list either the memory or the contexts currently in use.

Syntax: **list** all

contexts usage

memory usage

all Displays the contexts in use and the interfaces using the contexts, and the memory usage statistics. The output is a combination of list contexts usage and list memory usage displays.

Example: `list all`

context usage

Displays all of the compression contexts currently allocated by an interface. This display allows you to see which interfaces are currently compressing data traffic

Example: `list context usage`

```
Compression System Context (Data Dictionary) Usage
--------------------------------------------------

    CTX  Net Interface   Channel   Status
    ---  --- ----------  --------- ------------------
      0    2  FR/0              16 In use
      1    1  PPP/0              1 In use

Total: 10     Free: 8     In Use/Reserved: 2
```

CTX This is the context number, which is an identifying tag for the context. The device creates a pool of contexts when it boots, and assigns a number to each context in the pool. The context number is also displayed in some of the compression-related ELS messages.

Net This is the number of the network interface which has allocated a particular context.

Interface This is the name of the network interface.

Channel The channel is an identifier used to distinguish between multiple contexts allocated to the same network interface. The network number and channel number together uniquely identify a single compression stream. For PPP links, only a single compressed data stream runs on the link, and this number will always be 1. For Frame Relay links, this number is the virtual circuit number (DLCI) of the particular circuit that is carrying compressed traffic.

Status This field indicates the current status of the context, which will almost always be “In use.” Occasionally “Defunct” may appear which indicates that compression has been shut down on a link, but that the context has not yet been released to the pool for reuse.

memory usage

Displays basic statistics about the current state of the compression feature. The output shows the number of compression contexts which have been allocated, the number of contexts currently in use, the amount of memory required by a context, and the total amount of memory reserved for compression contexts.

Example: `list memory usage`

```
Compression System Memory Usage Statistics
-------------------------------------------

Number of contexts allocated:           10         in use: 2
Size of compression context:            16948
   = Max compression history size:      12496
   + Max decompression history size:    4424
   + Overhead:                          28
Total memory allocated for contexts:    169480
```

Using Data Compression on PPP Links

The 2210 uses the PPP Compression Control Protocol (CCP) to negotiate the use of compression on a link. CCP provides a generalized mechanism to negotiate the use of a particular compression protocol, possibly even using a different protocol in each direction of the link, and various protocol-specific options. The software supports the Stac-LZS and MPPC protocols, so the peer must also provide support for at least one of these algorithms to successfully negotiate data compression between the two nodes. The two nodes must also agree on the algorithm-specific options for compression to operate.

Configuring Data Compression on PPP Links

To configure data compression on PPP links:

1. Enable the CCP protocol on the link with the **enable ccp** command. This enables the link to negotiate compression with the other node. Negotiation includes what compression protocol to use and any protocol-specific options.
2. Select which compression protocols may be negotiated using the **set ccp protocols** command.
3. Set the negotiable parameters for each compression protocol using the **set ccp options** command.

You can display the current compression configuration using the **list ccp** command.

Table 19-3 lists the available commands and Figure 19-3 on page 19-11 is an example of configuring compression on a PPP link. For detailed description of these commands, see “Point-to-Point Configuration Commands” on page 33-16.

Table 19-3. PPP Data Compression Configuration Commands

Data Compression Command	Action
disable ccp	Disables data compression.
enable ccp	Enables data compression.
set ccp options	Sets options for the compression algorithm.
set ccp algorithms	Specifies a prioritized list of compression protocols.
list ccp	Displays compression configuration.

```
Config> network 1 1

Point-to-Point user configuration
PPP Config> enable ccp
PPP Config> set ccp options 2
STAC: # histories [1]? 1
STAC: check mode (0=none, 1=LCB, 2=CRC, 3=Seq, 4=Ext) [3]? 3
PPP Config> list ccp
CCP Options
-----------

Data Compression enabled
Algorithm list: STAC-LZS
Stac: histories 1
Stac: check_mode SEQ
```

Figure 19-3. Example of Configuring Compression on a PPP Link

Notes:

1. The network command selects the network interface for the PPP link. If the link is a PPP dial circuit, you must then use the **encapsulator** command to access the PPP configuration menu.
2. If you enable CCP and do not set protocols for the link, the software automatically sets the link to use protocols STAC and MPPC as if you had entered the command **set ccp protocols stac mppc**.

   If you set multiple protocols, the order of the protocols determines the negotiation preference for the link.

   Certain dial-in client implementations may not be able to connect if the router supports multiple compression protocols on one link. If you encounter this, set the ccp protocol to either STAC or MPPC.
3. If you enter **set ccp protocols none**, the software will automatically disable compression on the link.

Monitoring Compression on PPP Links

You monitor compression as you would other PPP components. “Accessing the Interface Console Process” on page 34-1 describes how to access the PPP console environment and details about the commands. Table 19-4 lists the compression-related commands. Figure 19-4 on page 19-12 shows an example of listing compression on a PPP interface.

Table 19-4. PPP Data Compression Monitoring Commands

Command	Function
list control ccp	Lists CCP state and negotiated options.
list ccp	Lists CCP packet statistics.
list cdp or **list compression**	Lists compressed datagram statistics.

```
+ network 1
PPP > list control ccp

CCP State:                Open
Previous State:           Ack Sent
Time Since Change:        2 minutes and 52 seconds

Compressor:   STAC-LZS histories 1, check_mode SEQ
Decompressor: STAC-LZS histories 1, check_mode SEQ

PPP > list ccp

CCP Statistic             In                        Out
-------------             --                        ---

Packets:                  2                         3
Octets:                   18                        27
Reset Reqs:               0                         0
Reset Acks:               0                         0
Prot Rejects:             1                         -

PPP > list cdp

Compression Statistic     In                        Out
---------------------     --                        ---

Packets:                  19541                     19542
Octets:                   2550673                   2740593
Compressed Octets:        821671                    899446
Incompressible Packets:   0                         0
Discarded Packets:        0                         -
Prot Rejects:             0                         -
Compression Ratios:       3.11                      3.24
```

Figure 19-4. Monitoring Compression on a PPP Interface

Using Data Compression on Frame Relay Links

After configuring the global compression parameters and enabling compression on the interface, you must then set the parameters for each individual circuit (PVC) on the Frame Relay interface. Each circuit defined for the interface may have compression enabled on the circuit, and each circuit which successfully negotiates the use of compression uses one compression context from the global pool. You can also disable compression on the interface which means none of the circuits on that interface will be eligible to carry compressed data traffic.

Configuring Data Compression on Frame Relay Links

To configure data compression on FR links:

1. Enable compression on the interface using the **enable compression** command. This enables the link to negotiate compression with the other node.
2. Enable compression on each new PVC that will carry compressed data with the **add permanent-virtual-circuit** command. You can change existing PVCs using the **change permanent-virtual-circuit** command.

You can display the current compression configuration using the **list lmi** or **list permanent-virtual-circuit** commands.

Table 19-5 on page 19-14 lists the commands available for configuring compression on a Frame Relay link and Figure 19-5 on page 19-13 is an example of configuring a Frame Relay Link. See “Frame Relay Configuration Commands” on page 31-16 for details about the Frame Relay configuration commands.

```
Config> net 2

Frame Relay user configuration

FR Config> enable compression
Maximum number of run-time compression PVCs (zero means no limit) [0]? 0
Do you want orphan PVCs to perform compression [Y]? n
The number of currently defined non-compression PVCs is 4
Would you like to change them all to compression PVCs [N]? y

FR Config> add perm

Circuit number [16]? 22
Committed Information Rate (CIR) in bps [65536]?
Committed Burst Size (Bc) in bits [64000]?
Excess Burst Size (Be) in bits [0]?
Assign circuit name []? cir22
Is circuit required for interface operation [N]?
Do you want to have data compression performed [Y]?

FR Config>list lmi

                        Frame Relay Configuration


  LMI enabled            =      No  LMI DLCI                      =       0
  LMI type               =    ANSI  LMI Orphans OK                =     Yes
  CLLM enabled           =      No  Timer Ty seconds              =      11

  Protocol broadcast     =     Yes  Congestion monitoring         =     Yes
  Emulate multicast      =     Yes  CIR monitoring                =      No
  Notify FECN source     =      No  Throttle transmit on FECN     =      No

  Data compression       =     Yes  Orphan compression            =      No
  Compression PVC limit  =    None  Number of compression PVCs    =       2

  PVCs P1 allowed        =      64  Interface down if no PVCs     =      No
  Timer T1 seconds       =      10  Counter N1 increments         =       6
  LMI N2 error threshold =       3  LMI N3 error threshold window =       4
  MIR % of CIR           =      25  IR % Increment                =      12
  IR % Decrement         =      25  DECnet length field          =      No
  Default CIR            =   65536  Default Burst Size            =   64000
  Default Excess Burst   =       0

FR Config>list perm

  Maximum PVCs allowable =      64
  Total PVCs configured  =       2

            Circuit               Circuit   Circuit     CIR    Burst  Excess
             Name                 Number     Type      in bps  Size   Burst
--------------------------------- ------- ----------- ------- ------- -----
 circ16                             16    @ Permanent   65536   64000       0
 cir22                              22    @ Permanent   65536   64000       0

* = circuit is required
# = circuit is required and belongs to a required PVC group
@ = circuit is data compression capable
```

Figure 19-5. Example of Configuring Compression on a Frame Relay Link

Table 19-5. Data Compression Configuration Commands

Command	Action
add permanent-virtual-circuit *#*	Use to enable data compression on a specific PVC defined on an interface.
change permanent-virtual-circuit *#*	Use to change whether a specific PVC will compress data.
disable compression	Disables data compression.
enable compression	Enables data compression.
list lmi	Displays the current configuration of the interface.
list permanent	Lists summary information about circuits.

Note: Enabling compression on orphan circuits will decrease the number of available compression contexts available for the native PVCs on the device.

If you enable compression on a Frame Relay interface, that already has compression enabled, the software asks you if you want to change compression parameters on the interface as shown in 19-14. You can change compression on the interface without disabling compression.

```
Example of changing compression on Frame Relay Interfaces
Config> net 2

Frame Relay user configuration

FR Config> enable compression
Data compression already enabled.
Do you wish to continue and change an interface parameter [Y]
Maximum number of run-time compression PVCs (zero means no limit) [0]? 32
Do you want orphan circuits to perform compression []?
Do you want to change the compression capability of all of your existing PVCs [N]?
```

Monitoring Data Compression on Frame Relay Links

You monitor compression as you would other Frame Relay components. “Frame Relay Console Commands” on page 32-1 describes how to access the Frame Relay console environment and details about the commands. Table 19-6 lists the compression-related commands. Figure 19-6 on page 19-15 shows an example of listing compression on a Frame Relay interface.

Table 19-6. Frame Relay Data Compression Monitoring Commands

Command	Display
list lmi	Lists the current status of the interface.
list permanent	Lists summary information about circuits.
list circuit	Lists the current status of a circuit.

```
+ network 2
FR 2 > list lmi

Management Status:
  ------------------

    LMI enabled          =      No  LMI DLCI                   =       0
    LMI type             =    ANSI  LMI Orphans OK             =     Yes
    CLLM enabled         =      No

    Protocol broadcast   =     Yes  Congestion monitoring      =     Yes
    Emulate multicast    =     Yes  CIR monitoring             =      No
    Notify FECN source   =      No  Throttle transmit on FECN  =      No
    PVCs P1 allowed      =      64  Interface down if no PVCs  =      No
    Line speed (bps)     =   64000  Interface MTU in bytes     =    2048
    Timer T1 seconds     =      10  Counter N1 increments      =       6
    LMI N2 threshold     =       3  LMI N3 threshold window    =       4
    MIR % of CIR         =      25  IR % Increment             =      12
    IR % Decrement       =      25  DECnet length field        =      No
    Default CIR          =   65536  Default Burst Size         =   64000
    Default Excess Burst =       0

    Current receive sequence  =       0
    Current transmit sequence =       0
    Total status enquiries    =       0  Total status responses   =       0
    Total sequence requests   =       0  Total responses          =       0

    Data compression enabled  =     Yes  Orphan Compression       =      No
    Compression PVC limit     =    None  Active compression PVCs  =       1

  PVC Status:
  -----------

    Total allowed  =      64  Total configured  =       1
    Total active   =       1  Total congested   =       0
    Total left net =       0  Total join net    =       0

FR 2 > list permanent

 Circuit                                 Orphan  Type/   Frames      Frames
 Number           Circuit Name           Circuit State Transmitted  Received
 ------- ------------------------------- ------- ----- ----------- --------

     16  circ16                            No   @ P/A        58364      58355
     22  circ22                            No   & P/A        58364      58355

    A - Active   I - Inactive   R - Removed   P - Permanent   C - Congested
    * - Required               # - Required and belongs to a PVC group
    @ - Data compression capable but not operational
    & - Data compression capable and operational
```

Figure 19-6 (Part 1 of 2). Monitoring Compression on a Frame Relay Interface or Circuit

```
FR 2 > list circuit 22

 Circuit name = circ22

     Circuit state        =    Active  Circuit is orphan   =           No
     Frames transmitted   =     58391  Bytes transmitted   =      2676894
     Frames received      =     58383  Bytes received      =      2671009
     Total FECNs          =         0  Total BECNs         =            0
     Times congested      =         0  Times Inactive      =            0
     CIR in bits/second   =     65536  Potential Info Rate =        64000
     Committed Burst (Bc) =     64000  Excess Burst (Be)   =            0
     Minimum Info Rate    =     16000  Maximum Info Rate   =        64000
     Required             =        No  PVC group name      =   Unassigned

     Compression capable  =       Yes  Operational         =          Yes
     R-R's received       =         0  R-R's transmitted   =            0
     R-A's received       =         0  R-A's transmitted   =            0
     R-R mode discards    =         0  Enlarged frames     =            0
     Decompress discards  =         0  Compression errors  =            0
     Rcv error discards   =         0

     Compression ratio    =  1.00 to 1  Decompression ratio =    1.00 to 1

     Current number of xmit frames queued                  =            0
     Xmit frames dropped due to queue overflow             =            0
```

Figure 19-6 (Part 2 of 2). Monitoring Compression on a Frame Relay Interface or Circuit

Chapter 20. Configuring Local or Remote Authentication

Authentication is the action of determining who a user (or entity) is. Authenticating user access for the PPP protocol on the 2210 extends the flexibility of user profile management as it relates to PPP authentication protocols PAP, CHAP, and SPAP . See “PPP Authentication Protocols” on page 33-7 for additional information about configuring PAP , CHAP, and SPAP .

Authentication can be configured locally or can be configured to consolidate user configuration by using authentication servers that are available on the network to service authentication requests for the entire network. The IBM 2210 implements locally maintained authentication as well as the following authentication server protocols:

- Radius
- TACACS
- TACACS+

Understanding Authentication Servers

An ***authentication server*** is a server in the network that validates userids and passwords for the network. If a device is configured for authentication through an authentication server and the device receives a packet from an authentication protocol, the device passes a userid and password to the server for authentication. If the userid and password are correct, the server responds positively. The device can then communicate with the originator of the request. If the server does not find the userid and password it receives from the device, it responds negatively to the device. The device then rejects the session from which it got the authentication request.

Accessing the Authentication Configuration Prompt

To access the `Authent config >` prompt:

1. Enter **talk 6** at the `*` prompt.
2. Enter **feature auth** at the `Config >` prompt.

Authentication Configuration Commands

Table 20-1 lists the commands available at the `Authent config >` prompt.

Table 20-1. Authentication Configuration Commands

Command	Function
? (Help)	Displays all the Authentication commands or the options available for a specific command.
List	Displays the Authentication configuration parameters.
Set	Configures Authentication parameters.
ppp-users	Configures ppp-user parameters.
Quickset	Configures the authentication method quickly.
Exit	Returns you to the previous prompt level.

? (Help)

Displays the Authentication commands or lists parameters for specific Authentication commands.

Syntax: ?

Example: ?

```
LIST
SET
PPP-USERS
QUICKSET
EXIT
```

Example: set ?

```
AUTHENTICATION-TYPE
DEFAULT-USER
```

Example: set aut ?

```
LOCAL
RADIUS
TACACS
TACACSPLUS
```

SET

Use the **set** command to set parameter values for the authentication protocol currently selected.

set authentication-type
default-user

Syntax: set authentication-type

Example: set authentication-type

local Sets the authentication type to use a locally-maintained user database.

radius Sets the authentication type to use the radius authentication server protocol.

Values for the following parameters can be set:

key-for-encryption:
Specifies the encryption key.

Valid Values: Any alphanumeric character string up to 32 characters long.

Default Value: None.

primary-server-address:
Specifies the address of the primary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

retry-parameters

Valid Values:

Default Value:

secondary-server-address:
Specifies the address of the secondary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

tacacs Sets the authentication type to use the TACACS authentication server protocol.

Values for the following parameters can be set:

primary-server-address:
Specifies the address of the primary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

retry-parmeters

Valid Values:

Default Value:

secondary-server-address:
Specifies the address of the secondary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

tacacsplus Sets the authentication type to use the TACACS+ authentication server protocol.

Values for the following parameters can be set:

encryption:
Specifies whether encryption will be used

Valid Values: yes or no

Default Value:

key-for-encryption:
Specifies the encryption key to be used

Valid Values: Any 16-hexadecimal digit value

Default Value:

primary-server-address:
Specifies the address of the primary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

privilege-level

Valid Values: 0 through 15

Default Value: 0

restarts
Sets the number of restarts. This parameter does not include timeout restarts and only pertains to restarts requested by the server.

Valid Values: 0 to 3200

Default Value: 0

time-to-connect
The amount of time to allow to obtain the authentication from the server.

Valid Values: 1 – 60

Default Value: 9

secondary-server-address:
Specifies the address of the secondary authentication server.

Valid Values: Any valid IP address

Default Value: 0.0.0.0

Syntax: set default-user

Example: `set default-user`

PPP-USERS

Use the **ppp-users** command to configure user profiles in the locally administered database. This command is similar to the ppp-user functions available in the PPP configuration environment, but only uses single commands such as add or change. See Chapter 33, “Using and Configuring Point-to-Point Protocol Interfaces” on page 33-1 for details of the PPP commands that you use in this environment.

PPP-USERS Command	Related PPP Command
add	add ppp_user
change	change ppp_user

Syntax: ppp-users

Example: `ppp-users`

Quickset

Use the **quickset** command to configure authentication quickly. The software prompts you for all the required information in a step-by-step manner.

Syntax: quickset

Example: `quickset`

List

Use the **list** command to display currently configured authentication information. The information provided depends on the Authentication type currently configured.

Syntax: list

Exit

Use the **exit** command to return to the previous command prompt.

Syntax: exit

Example: `exit`

Chapter 21. Getting Started with Network Interfaces

The chapters of this book describe how to configure and monitor network interfaces and link layer protocols supported by the Router. The purpose of this chapter is to give you some basic configuration and monitoring guidelines. This chapter also provides you with basic procedures and information needed for monitoring the interfaces via the GWCON **interface** command. Sections in this chapter include:

- "Before You Continue"
- "Network Interfaces and the GWCON Interface Command"
- "Accessing Network Interface Configuration and Console Processes"
- "Accessing Link Layer Protocol Configuration and Console Processes"
- "Defining Spare Interfaces" on page 21-2

Before You Continue

Before you continue, make sure that you have familiarized yourself with the procedures necessary for accessing the network interface configuration processes.

For more information on these procedures, refer to the sections that follow in this chapter.

Network Interfaces and the GWCON Interface Command

When configuring network interfaces, you may find it necessary to display certain information about specific interfaces. While some interfaces have their own console processes for monitoring purposes, the router displays statistics for *all* installed network interfaces when you use the **interface** command from the GWCON environment. (Refer to "Interface" on page 6-11.)

Accessing Network Interface Configuration and Console Processes

The follow references contain the background information and examples of how to access the configuration and console prompts for interfaces.

Refer to "Accessing Network Interface Configuration and Console Processes" on page 1-17, "Accessing the Network Interface Configuration Process" on page 1-18, and "Accessing the Network Interface Console Process" on page 1-21 for complete information on accessing interface configuration and console processes. Accessing these processes allows you to change and monitor software configurable parameters for network interfaces used in your router.

Accessing Link Layer Protocol Configuration and Console Processes

Refer to Chapter 1, "Getting Started (Introduction to the User Interface)" on page 1-1 for complete information on accessing the protocol configuration and console processes. Accessing these processes allows you to change and monitor configurable parameters for Link Layer protocols supported by your router. :subject

Defining Spare Interfaces

There may be occasions when you will need to define interfaces on your device that do not currently exist. You accomplish this ***dynamic reconfiguration*** of a device by defining spare interfaces while you are configuring the device and then using the console process to activate the interfaces when they are present. See “Configuring Spare Interfaces” on page 3-7 and “Activate” on page 6-4 for details.

Chapter 22. Configuring IEEE 802.5 Token-Ring Network Interfaces

This chapter describes how to set software configurable information for token-ring interfaces in the router. It includes the following sections:

- "Accessing the Interface Configuration Process"
- "Token-Ring Configuration Commands"

Accessing the Interface Configuration Process

To display the `TKR config>` prompt, enter the network command followed by the interface number of the Token-Ring interface. For example:

```
Config>network 0
Token-Ring interface configuration
TKR Config>
```

Use the **list devices** command at the `Config>` prompt to display a list of interface numbers configured on the router.

Note: Whenever you change a parameter, you must restart the router for the changes to take effect.

Token-Ring Configuration Commands

This section summarizes and then explains the token-ring configuration commands. Enter the commands at the `TKR config>` prompt. Table 22-1 lists token-ring configuration commands.

Table 22-1. Token-Ring Configuration Command Summary

Command	Function
? (Help)	Displays all the token-ring commands or lists subcommand options for specific commands.
Frame	Sets the NetWare IPX encapsulation type.
List	Displays the selected token-ring interface configuration.
LLC	Accesses the LLC configuration environment and subcommands.
Media	Sets the media-type as shielded or unshielded.
Packet-size	Changes packet-size defaults for all token-ring networks.
Set	Sets the aging timer for the RIF cache and the physical (MAC) address.
Source-routing	Enables or disables source-routing on the interface.
Speed	Sets the interface speed in Mbps.
Exit	Exits the token-ring configuration process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

```
EXIT
FRAME
LIST
LLC
MEDIA
PACKET-SIZE
SET
SOURCE-ROUTING
SPEED Mb/sec

media ?
```

Frame

Use the **frame** command to set the NetWare IPX encapsulation type. Enter one of the following:

Option	Description	Syntax
Token-Ring using MSB	Uses the standard 802.2 IPX header with the non-canonical Token-Ring address bit ordering (MSB).	frame token-ring msb
Token-Ring using LSB	Uses the 802.2 IPX header with the canonical address bit ordering (LSB).	frame token-ring lsb
Token-Ring with 802.2 SNAP using MSB	Uses the 802.2 format with a SNAP header and non-canonical address bit ordering. This encapsulation is used primarily in bridging environments.	frame token-ring_snap msb
Token-Ring with 802.2 SNAP using LSB	Uses the 802.2 format with a SNAP header and canonical address bit ordering.	frame token-ring_snap lsb

Syntax: frame *encapsulation type*

Example: **frame token_ring msb**

Note: The **frame** command cannot be used in a network configuration process to set an encapsulation until the interface has been properly configured via the IPX configuration process.

List

Use the **list** command to display the current configuration for the token-ring interface.

Note: If the MAC address is 0, the default station address is used.

Syntax: list

Example: **`list`**

```
Token-Ring configuration:

    Packet size (INFO field): 2052
Speed:                    16 Mb/sec
Media:                    Shielded

RIF Aging Timer:          120
Source Routing:           Enabled
MAC Address:              000000000000
NetWare IPX encapsulation: TOKEN-RING MSB
```

Packet size	Size of the token-ring packet.
Speed	Speed of the network.
Media	Type of media the network uses, shielded or unshielded.
RIF Aging Timer	Amount of time that the router holds the information contained in the Routing Information Field (RIF).
Source Routing	Status of the source-routing feature, enabled or disabled.
MAC Address	Configured MAC address that was set with the **set physical-address** command. If all zeros are displayed, the MAC address is the default address.
Netware IPX encapsulation	Configured NetWare IPX encapsulation type that was set with the **frame** command.

LLC

Use the **LLC** command to access the LLC configuration environment. See “LLC Configuration Commands” on page 24-1 for an explanation of each of these commands.

Syntax: llc

Example: **`llc`**

```
LLC config>
```

Note: If APPN is not included in your router software load, you will receive the following message if you try to use this command:

```
LLC configuration is not available for this network.
```

The LLC configuration environment is only available if APPN is included in the software load.

Media

Use the **media** command to change the network media type. The default media type is STP cable. Valid media type values are shielded and unshielded. Enter the media command followed by the *media-type*.

Syntax: media *media-type*

Example: **`media unshielded`**

Packet-Size

Use the **packet-size** command to change packet-size defaults for all token-ring networks. Enter the **packet-size** command followed by the desired number of bytes.

Table 22-2. Token-Ring 4/16 Valid Packet Sizes

Network Data Speed	Values (# of bytes)
4 Mbps	2052, 4399 **Note:** If you configure a packet size greater than 4399, the software sets the packet size to 4399.
16 Mbps	1470, 2052, 4399, 8130, 11407, and 17749

Note: If packet sizes are increased, buffer memory requirements will also increase.

Syntax: packet-size *#bytes*

Example: **`packet-size 4399`**

Set

Use the **set** command to set the Routing Information Field (RIF) timer and the physical (MAC) address.

Syntax: set physical-address
rif-timer

physical-address
Indicates whether you want to define a locally administered address for the Token-Ring interface's MAC sublayer address, or use the default factory station address (indicated by all zeroes). The MAC sublayer address is the address that the Token-Ring interface uses to receive and transmit frames.

Note: Pressing **Return** leaves the value the same. Entering **0** and pressing **Return** causes the router to use the factory station address. The default is to use the factory station address.

Valid values: Any 12-digit hexadecimal address.

Default value: burned-in address (indicated by all zeroes).

Example: **`set physical-address`**

```
MAC address in 00:00:00:00:00:00 form []?
```

rif-timer
Sets the maximum amount of time (in seconds) that the information in the RIF is maintained before it is refreshed. The default is 120.

Example: **`set rif-timer`**

```
RIF aging timer value [120]?  120
```

Source-routing

Use the **source-routing** command to enable or disable end station source routing. Source routing is the process by which end stations determine the source route to use to cross source routing bridges. Source routing allows the IP, IPX, and AppleTalk Phase 2 protocols to reach nodes on the other side of the source routing bridge.

This switch is completely independent of whether this interface is providing source routing via the SRT forwarder. The default setting is enabled.

Some stations cannot properly receive frames with a Source Routing RIF on them. This is especially common among NetWare drivers. Disabling source routing in this situation will allow you to communicate with these stations.

Source routing should be enabled only if there are source-routing bridges on this ring that you want to bridge IP, IPX, and AppleTalk Phase 2 packets through. Source routing must also be enabled so LLC test response messages can be returned.

Syntax: source-routing enable
disable

Example: **`source-routing enable`**

Speed

Use the **speed** command to change data speed. The default speed is 4 Mbps. Enter the **speed** command followed by the speed-value (in Mbps).

Syntax: speed *speed-value*

Example: **`speed 16`**

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

Chapter 23. Monitoring IEEE 802.5 Token-Ring Network Interfaces

This chapter describes how to monitor specific Token-Ring interfaces in the router by using either the interface console commands or the GWCON interface command. It includes the following sections:

Accessing the Interface Console Process

To display the token-ring monitoring prompt (TKR>), enter the network command followed by the interface number of the Token-Ring interface. For example:

```
+network 0
TKR>
```

Use the **list devices** command at the Config> prompt to display a list of interface numbers configured on the router.

Follow the procedure described in Chapter 21, "Getting Started with Network Interfaces" on page 21-1 to access the interface console process for the interface described in this chapter. Once you have accessed the desired interface console process, you can begin entering console commands.

Token-Ring Interface Console Commands

This section summarizes and explains the Token-Ring console commands. Enter commands at the TKR> monitoring prompt. Table 23-1 lists the console commands.

Table 23-1. Token-Ring Console Command Summary

Command	Function
? (Help)	Displays all the Token-Ring commands or lists subcommand options for specific commands.
Dump	Displays a dump of the RIF cache.
LLC	Displays the LLC console monitoring prompt.
Exit	Exits the Token-Ring console process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

```
dump
llc
exit
```

Dump

When source routing is enabled in the `tkr config>` process, you can use the **dump** command to request a dump of the RIF cache contents.

Syntax: dump

Example: **`dump`**

```
MAC address    State     Usage   RIF
0000C90B1A57   ON_RING   Yes     0220
```

MAC address — Displays the MAC address of the Token-Ring interface.

State — Displays one of the interface states:

On_ring - indicates that a RIF was found for a node on the ring.
Have_route - indicates that a RIF was found for a node on a remote ring.
No_route - is displayed for a brief period of time as an explorer frame is sent out and the router is waiting for a return.
Discovering - indicates that the router sent an explorer frame to rediscover the RIF.
St_route - indicates that a route obtained from a Spanning tree explorer.

Usage — Indicates that a RIF was used in a packet. The number is arbitrary and has no functional significance.

RIF — Displays a code that indicates the RIF in hexadecimal.

Note: The RIF is displayed only if Source Route Bridging is enabled on the token-ring interface.

- NetBIOS RIF data can be displayed using the following sequence of commands: **talk 5, protocol ASRT, name-caching, list cache rifs.**
- Data Link Switching RIF data can be displayed using the following sequence of commands: **talk 5, protocol dlsw, list llc2 session all.**

LLC

Use the **LLC** command to access the LLC monitoring prompt. LLC commands are entered at this new prompt. See “LLC Monitoring Commands” on page 25-1 for an explanation of each of these commands.

Syntax: llc

Example: **`llc`**

```
LLC user monitoring
LLC>
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

Token-Ring Interfaces and the GWCON Interface Command

While Token-Ring interfaces have their own console processes for monitoring purposes, the router also displays complete statistics for installed network interfaces when you use the **interface** command from the GWCON environment.

Statistics Displayed for 802.5 Token-Ring Interfaces

The following statistics display when you enter the **interface** <***net #***> command for a Token-Ring interface from the GWCON environment.

```
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed

0  0   TKR/0      6000000   1C          1          0            0
Token-Ring/802.5  MAC/data-link on IBM Token-Ring interface
Microcode version: 000VL00A0 (050394)

Physical address        000C90820C7
Network speed           16 Mbps
Max packet size (INFO)  2052
Handler state           Ring open
Ring status             SERR | CO
Interface Restarts      0

# times Signal lost     0          # times Beaconing  0
Hard errors             0          Lobe wire faults   0
Auto-removal errors     0          Removes received   0
Ring recovery actions   0

Line errors             0          Burst errors       0
ARI/FCI errors          0          Inputs dropped     0
Frame copy errors       0          Token errors       0
Lost frames             0
```

The following section describes general interface statistics:

Nt	Global interface number
Nt'	Applies only to dial circuits
Interface	Interface name and Number of this interface within interfaces of type "intrfc"
CSR	COMM and Status Registers address
Vec	Interrupt vector
Self-Test: Pass	Number of times self-test succeeded
Self-Test: Fail	Number of times self-test failed
Maint: Fail	Number of maintenance failures

The following section describes the statistics displayed that are specific to the Token-Ring interfaces:

Physical address	Specifies the physical address of the Token-Ring interface.
Network speed	Specifies the speed of the Token-Ring network that connects to the interface. The Network Speed counter displays the number of packets that the interface can pass per second.

Max packet size (info)	Displays the maximum packet size configured for that interface. The Max Packet Size counter displays the maximum length, in bytes, of a packet that the interface transmits or receives. This counter is user-defined.
Handler state	Displays the current state of the Token-Ring handler. The Handler state counter displays the state of the handler after the self-test runs.
Ring status	Last Ring Status of the Token Ring interface.

SIGL	SIGNAL_LOSS The interface has detected a loss of signal on the ring.
HERR	HARD_ERROR The interface is presently transmitting or receiving beacon frames on the ring.
SERR	SOFT_ERROR The interface has transmitted a report error MAC frame.
BEAC	TRANSMIT_BEACON The interface is transmitting beacon frames to or from the ring.
LWF	LOBE_WIRE_FAULT The interface has detected an open or short circuit in the cable between the interface and the wiring concentrator. The interface is closed and is at the state following initialization.
ARMV	AUTO_REMOVAL_ERROR The interface has failed the lobe wrap test, which resulted from the beacon auto-removal process, and has removed itself from the ring. The interface has closed and is at the state following initialization.
RMVD	REMOVED_RECEIVED The interface has received a remove ring station MAC frame request and has removed itself from the ring. The interface is closed and is at the state following initialization.
CO	COUNTER_OVERFLOW One of the following error counters has incremented from 254 to 255: Line, ARI/FCI, Frame Copy, Lost Frames, Burst, Lobe wire faults, Removes received. This display shows these error counters.
SSTA	SINGLE_STATION The interface has sensed that it is the only station on the ring.
RR	RING_RECOVERY The interface observes claim Token MAC frames on the ring. The interface may be transmitting the claim Token frames. This status remains until the interface transmits a ring purge frame.

Interface Restarts	Specifies the number of times the Token Ring chip timed out, or the Token Ring driver received a bad command from the handler. For information about why a restart occurred, see messages TKR.37, TKR.38, TKR.39, TKR.40, and TKR.41. in *Event Logging System Messages Guide*
# of times signal lost	Specifies the total number of times that the router was unable to transmit a packet due to loss of signal.
Hard errors	Displays the number of times the interface transmits or receives beacon frames from the network.
Auto-removal errors	Displays the number of times the interface, due to the beacon auto-removal process, fails the lobe wrap test and removes itself from the network.
Ring recovery actions	Displays the number of times the interface detects claim token medium access control (MAC) frames on the network.

Line errors	The Line Errors counter increments when a frame is repeated or copied and the Error Detected Indicator (EDI) is zero for the incoming frame: One of the following conditions must also exist: • A token with a code violation exists. • A frame has a code violation between the starting and ending delimiter. • A Frame Check Sequence (FCS) error occurs.
ARI/FCI errors	The ARI/FCI (Address Recognized Indicator/Frame Copied Indicator) Errors counter increments if the interface receives either of the following: An Active Monitor Present (AMP) MAC frame with the ARI/FCI bits equal to zero and a Standby Monitor Present (SMP) MAC frame with the ARI/FCI bits equal to zero. More than one SMP MAC frame with the ARI/FCI bits equal to zero, without an intervening AMP MAC frame. This error indicates that the upstream neighbor copied the frame but is unable to set the ARI/FCI bits.
Frame copy errors	Displays the number of times the interface in receive/repeat mode recognizes a frame addressed to its specific address but finds the address recognize indicator (ARI) bits not equal to zero. This error indicates a possible line hit or duplicate address.
Lost frames	Displays the number of times the interface is in transmit mode (stripping) and fails to receive the end of a transmitted frame.
# times beaconing	Displays the number of times the interface transmits a beacon frame to the network.
Lobe wire faults	Displays the number of times the network detects an open or short circuit in the cable between the interface and the wiring concentrator.
Removes received	Displays the number of times the interface receives a remove ring station MAC frame request and removes itself from the network.
Burst errors	Displays the number of times the interface detects the absence of transitions for five half-bit times between the start delimiter (SDEL) and the end delimiter (EDEL) or between the EDEL and the SDEL.
Inputs dropped	Displays the number of times an interface in repeat mode recognizes a frame addressed to it but has no buffer space available to copy the frame.
Token errors	The token errors counter increments when the active monitor detects a token protocol with any of the following errors: The MONITOR_COUNT bit of token with nonzero priority equals one. The MONITOR_COUNT bit of a frame equals one. No token or frame is received within a 10-ms window. The starting delimiter/token sequence has a code violation in an area where code violations must not exist.

Chapter 24. Configuring LLC Interfaces

This chapter describes how to set software configurable information for Logical Link Control (LLC) interfaces in the router.

Logical Link Control can be thought of as a “sub-protocol.” It is not accessed directly from either the Talk 6 (configuration) or the Talk 5 (console) environment. Instead, it is accessed from the Token Ring, Point-to-Point (PPP), or Frame Relay protocols by entering an **LLC** command.

This chapter contains the following sections:

- “Accessing the Interface Configuration Process”
- “LLC Configuration Commands”

Accessing the Interface Configuration Process

Access the configuration commands for the protocol you wish to configure over LLC:

- Token Ring, as described in Chapter 22, “Configuring IEEE 802.5 Token-Ring Network Interfaces” on page 22-1
- Point-to-Point, as described in Chapter 33, “Using and Configuring Point-to-Point Protocol Interfaces” on page 33-1
- Frame Relay, as described in Chapter 31, “Using and Configuring Frame Relay Interfaces” on page 31-1

Each of these prompt levels has an LLC command. Enter **LLC** to access the LLC configuration commands and perform LCC configuration. When you are finished, enter **Exit** to return to the prompt level for the protocol you are configuring.

LLC Configuration Commands

LLC configuration is required when you need to pass packets over an SNA network. To enter these commands, you must first enter the LLC configuration environment (see “Accessing the Interface Configuration Process” on page 22-1).

This section summarizes and then explains all of the LLC configuration commands. These commands (Table 24-1) enable you to configure LLC when you need to pass packets over a SNA network.

Table 24-1. LLC Configuration Command Summary

Command	Function
? (Help)	Displays all the LLC commands or lists subcommand options for specific commands.
List	Displays the selected LLC configuration.
Set	Sets the timers associated with LLC, and the size of the transmit and receive windows.
Exit	Exits the LLC configuration process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

```
list
set
exit
```

List

Use the **list** command to display the current configuration for the LLC.

Syntax: list

Example: **list**

```
Reply Timer (T1):                 1 seconds
Receive ACK Timer (T2):           100 milliseconds
Inactivity Timer (Ti):            30 seconds
Max Retry value (N2):             8
Rcvd I-frames before ACK (N3):    1
Transmit Window (Tw):             2
Receive Window (Rw):              2
Acks needed to increment Ww (Nw): 1
```

Reply Timer (T1)	This timer expires when the LLC fails to receive a required acknowledgment or response from the other LLC station.
Receive ACK Timer (T2)	This timer is used to delay sending of an acknowledgment for a received I-format frame.
Inactivity Timer (Ti)	This timer expires when the LLC does not receive a frame for a specified time period. When this timer expires the LLC transmits an RR until the other LLC responds or the N2 retry count is exceeded. Default is 30 seconds.
Max Retry value (N2)	The maximum number of retries by the LLC protocol. Default is 8.
Rcvd I-frames before ACK (N3)	This value is used with the T2 timer to reduce acknowledgment traffic for received I-frames. This counter sets a specified value and decrements each time an I-frame is received. When this counter reaches 0 or the T2 timer expires, an acknowledgment is sent. Default is 1.
Receive Window (Rw)	Indicates the maximum number of unacknowledged sequentially numbered I-frames that an LLC can receive from a remote host.
Transmit Window (Tw)	Indicates the maximum number of I-frames that can be sent before receiving an RR.
Acks needed to increment Ww (Nw)	This field is set to a default value of 1.

Set

Use the **set** command to configure the LLC.

Attention: Changing LLC parameters from the defaults can affect how the LLC protocol works.

Syntax: set n2-max-retry *count* n3-frames-rcvd-before-ack *count* nw-acks-to-inc-window *count* rw-receive-window *count* t1-reply-timer *seconds* t2-receive-ack-timer *seconds* ti-inactivity-timer *seconds* tw-transmit-window *count*

n2-max-retry

The maximum number of retries by LLC protocol. For example, N2 is the maximum number of times the LLC transmits an RR without receiving an acknowledgment when the inactivity timer expires. Default is 8. Minimum is 1. Maximum is 127.

Example: `set n2-max-retry`

```
Max Retry value (N2) [8]?
```

n3-frames_rcvd-before-ack

This value is used with the T2 timer to reduce acknowledgment traffic for received I-frames. Set this counter to a specified value. Each time an I-frame is received, this value decrements. When this counter reaches 0 or the T2 timer expires, an acknowledgment is sent. Default is 1. Minimum is 1. Maximum is 255.

Example: `set n3-frames_rcvd-before-ack`

```
Number I-frames received before sending ACK(N3) [1]?
```

rw-receive-window

Indicates the maximum number of unacknowledged sequentially numbered I-frames that an LLC can receive from a remote LLC peer. This value must be equal to or less than 127.

Example: `set rw-receive-window`

```
Receive Window (Rw), 127 Max. [2]?
```

nw-acks-to-inc-ww

This field is set to a default value of 1.

t1-reply-timer

This timer expires when the LLC fails to receive a required acknowledgment or response from the other LLC station. When this timer expires, an RR is sent with the poll bit set and T1 is started again. If the LLC receives no response after the configured maximum number of retries (N2), the link underneath is declared inoperative. Default is 1. Minimum is 1. Maximum is 256.

Example: `set t1-reply-timer`

```
Reply Timer (T1) in sec. [1]?
```

t2-receive-ack-timer

This timer is used to delay sending of an acknowledgment for a received I-format frame. This timer is started when an I-frame is received. The timer is reset when an acknowledgment is sent. If this timer expires, LLC2 sends an acknowledgment as soon as possible. Set this value so that it is less than that of T1. This insures that the remote LLC2 peer receives the delayed acknowledgment before the T1 timer expires. Default is 1 (100 ms). Minimum is 1. Maximum is 2560.

Example: `set t2-receive-ack-timer`

```
Receive Ack timer (T2) in 100 millisec. [1]?
```

Note: If this timer is set to 1 (the default) it will not run (for example, **n3-frames_rcvd-before-ack=1**).

ti-inactivity-timer

This timer expires when the LLC does not receive a frame for a specified time period. When this timer expires the LLC transmits an RR until the other LLC responds or the N2 retry count is exceeded. Default is 30 seconds. Minimum is 1 second. Maximum is 256 seconds.

Example: `set ti-inactivity-timer`

```
Inactivity Timer (Ti) in sec. [30]?
```

tw-transmit-window

Sets the maximum number of I-frames that can be sent before receiving an RR. Assuming that the other end of the LLC session can actually receive this many consecutive I-frames, and the router has enough heap memory to keep copies of these frames until an acknowledgment is received, increasing this value may increase the throughput. Default is 2. Minimum is 1. Maximum is 127.

Example: `set tw-transmit-window`

```
Transmit Window (Tw), 127 Max. [2]?
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

Chapter 25. Monitoring LLC Interfaces

This chapter describes how to monitor specific LLC interfaces in the router by using either the interface console commands or the GWCON interface command.

Logical Link Level can be thought of as a “sub-protocol.” It is not accessed directly from either the Talk 6 (configuration) or the Talk 5 (console) environment. Instead, it is accessed from the Token Ring, Point-to-Point (PPP), or Frame Relay protocols by entering an **LLC** command.

This chapter includes the following sections:

- “Accessing the Interface Console Process”
- “LLC Monitoring Commands”

Accessing the Interface Console Process

Access the console commands for the protocol you wish to monitor over LLC:

- Token Ring, as described in Chapter 23, “Monitoring IEEE 802.5 Token-Ring Network Interfaces” on page 23-1
- Point-to-Point, as described in Chapter 34, “Monitoring Point-to-Point Protocol Interfaces” on page 34-1
- Frame Relay, as described in Chapter 32, “Monitoring Frame Relay Interfaces” on page 32-1

Each of these prompt levels has an LLC command. Enter **LLC** to access the LLC console commands to monitor LCC. When you are finished, enter **Exit** to return to the prompt level for the protocol you are monitoring.

LLC Monitoring Commands

This section summarizes and then explains all of the LLC monitoring commands. These commands let you monitor the LLC while passing packets over an SNA network.

Table 25-1. LLC Monitoring Command Summary

Command	Function
? (Help)	Displays all the LLC commands or lists subcommand options for specific commands.
Clear-counters	Clears all statistical counters.
List	Displays interface, SAP, and session information.
Set	Allows the user to dynamically configure LLC parameters that are valid for the life of the session.
Exit	Exits the LLC monitoring process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

```
CLEAR-COUNTERS
LIST
SET
EXIT


set ?
```

Clear-Counters

Use the **clear-counters** command to clear all the LLC statistical counters.

Syntax: clear-counters

Example: **clear-counters**

List

Use the **list** command to display interface, service access point (SAP), and session information.

Syntax: list interface sap . . . session

interface
Displays all SAPs opened on this interface.

Example: **list interface**

```
SAP      Number of Sessions
F4       1
```

sap sap_number
Displays information for the specified SAP on the interface.

Example: **list sap**

```
SAP value in hex (0FE) [1]? F4

Interface                       0, TKR/0
Reply Timer(T1)                 1 sec
Receive ACK Timer (T2)          100 millisec
Inactivity Timer (Ti)           30 sec
MAX Retry Value (N2)            8
MAX I-field Size (N1)           2052
Rcvd I-frames before ACK (N3)   1
Transmit Window Size (Tw)       2
Acks Needed to Inc Ww (Nw)      1

Frame                           Xmt   Rcvd
UI-frames                       4     5
TEST-frames                     0     1
XID-frames                      0     0
I-frames                        291   26
RR-frames                       81    291
```

```
RNR-frames                      0     0
REJ-frames                      0     0
SABME-frames                    1     0
UA-frames                       0     1
DISC-frames                     0     0
DM-frames                       0     0
FRMR-frames                     0     0
I-frames discarded by LLC             0
I-frames Refused by LLC user          0

Cumulative number of sessions         1
Number of active sessions             1

Session ID                                              Remote
(int-sap-id)   Local MAC           Remote MAC             SAP   State
00F40000     00:00:C9:08:41:DB   10:00:5A:F1:02:37   F4    OPENED
```

SAP value in hex (0FE)	The SAP value of the session.
Interface	The interface number and type over which the session is running.
Reply Timer (T1)	Indicates the time it takes for this timer to expire when the LLC fails to receive an acknowledgment or response from the other LLC station.
Receive ACK Timer (T2)	Indicates the time delay the LLC uses before sending an acknowledgment for a received I-frame.
Inactivity Timer (Ti)	Indicates the time the LLC waits during inactivity before issuing an RR.
MAX Retry Value (N2)	The maximum number of retries by the LLC protocol.
MAX I-field Size (N1)	Maximum amount of data (in bytes) allowed in the I-field of an LLC2 frame.
Rcvd I-frame before ACK (N3)	Indicates the value that is used with T2 timer to reduce acknowledgment traffic for received I-frames.
Transmit Window Size (Tw)	Indicates the maximum number I-frames that can be sent before receiving an RR.
Acks Needed to Inc Ww (Nw)	This field is set to a default value of 1.
Frames Xmt and Rcvd	Counter that displays the total number of frame types transmitted (Xmt) and (Rcvd).
I-frames discarded by LLC	Counter that displays the total number of I-frames discarded by the LLC, usually because the sequence number is out of sequence.
I-frames refused by LLC user	Counter that displays the number of I-frames discarded by the software above the LLC. For example, DLSw (Data Link Switching).
Cumulative number of sessions	The total number of sessions that were opened over this SAP.
Number of active sessions	The total number of currently active sessions that are running over the interface.
Session ID (int-sap-id)	The session ID for the console interface.
Local MAC	The router's LLC MAC address.
Remote MAC	The remote LLC's MAC address.

Remote SAP	The remote SAP of the LLC connection.
Remote State	The finite state(s) that results from interaction between the LLC peers. There are 21 states that are described below.

Link_Closed	The remote LLC peer is not known to the local LLC peer and is considered as not existing.
Disconnected	The local LLC peer is known to the other peer. This LLC peer can send and receive XID, TEST, SABME, and DISC commands; and XID TEST, UA, and DM responses.
Link_Opening	The state of the local LLC peer after sending a SABME or UA in response to a received SABME.
Disconnecting	The state of the local LLC after sending a DISC command to the remote LLC peer.
FRMR_Sent	The local LLC peer has entered the frame reject exception state and has sent a FRMR response across the link.
Link_Opened	The local LLC peer is in the data transfer phase.
Local_Busy	The local LLC peer is unable to receive additional I-frames.
Rejection	A local LLC peer that has received one or more out-of-sequence I-frames.
Checkpointing	The local LLC peer has sent a poll to the remote LLC peer and is waiting for an appropriate response.
CKPT_LB	A combination of checkpointing and local busy states.
CKPT_REJ	A combination of the checkpointing and rejection states.
Resetting	The local LLC peer has received a SABME and is reestablishing the link.
Remote_Busy	The state that occurs when an RNR is received from the remote LLC peer.
LB_RB	A combination of local_busy and remote_busy states.
REJ_LB	A combination of rejection and local_busy states.
REJ_RB	A combination of rejection and remote_busy states.
CKPT_REJ_LB	A combination of checkpointing, rejection, and local_busy states.
CKPT_CLR	A combination state resulting from the termination of a local_busy condition while the LLC peer is CKPT_LB.
CKPT_REJ_CLR	A combination state resulting from the transfer of an unconfirmed local busy clear while the link station is in the CKPT_REJ_LB state.
REJ_LB_RB	A combination of the rejection, local_busy, and remote_busy states.
FRMR_Received	The local LLC peer has received an FRMR response from the remote LLC peer.

Session

Displays information on the specified LLC session that is open on the interface.

Example: `list session`

```
Session Id: [0]? 00-F4-0000

Interface0,                     TKR/0
Remote MAC addr                 10:00:5A:F1:02:37
Source MAC addr                 00:00:C9:08:35:47
Remote SAP                      F4
Local SAP                       F4
RIF                             (089E 0101 0022 0010)
Access Priority                 0
State                           LINK_OPENED
Replay Timer                    1 sec
Receive ACK Timer (T2)          100 millisec
Inactivity Timer (Ti)           30 sec
MAX I-field Size (N1)           2052
MAX Retry Value (N2)            8
Rcvd I-frames before ACK (N3)   1
Transmit Window Size (Tw)       2
Working Transmit Size (Ww)      2
Acks Needed to Inc Ww (Nw)      1
Current Send Seq (Vs)           9
Current Rcv Seq (Vr)            7
Last ACK'd sent frame (Va)      9
No. of frames in ACK pend q     0
No. of frames in Tx pend q      0
Local Busy                      NO
Remote Busy                     NO
Poll Retry count                8
Appl output flow stopped        NO
Send process running            YES

Frame                            Xmt   Rcvd
I-frames                         1456  2678
RR-frames                        502   403
RNR-frames                       0     0
REJ-frames                       0     0
I-frames discarded by LLC              0
I-frames Refused by LLC user           0
```

Session Id	Indicates the session ID number.
Interface	Indicates the number of the interface over which this session is running.
Remote MAC addr	Indicates the MAC address of the remote LLC peer.
Source MAC addr	Indicates the MAC address of the local LLC.
Remote SAP	The remote side SAP of the LLC connection.
Local SAP	The local side SAP of the LLC connection.
RIF	The actual RIF of the frame.
Access Priority	Priority of the packet. 07 for upper layer control.
State	The finite state(s) that results from interaction between the LLC peers. Refer to the **list sap** command on page 25-2 for more information.
Receive ACK timer (T2)	Indicates the time delay the LLC uses before sending an acknowledgment for a received I-frame.
Inactivity timer (Ti)	Indicates the time the LLC waits during inactivity before issuing an RR.

MAX I-field size (N1)	Maximum size of the data field (in bytes) of a frame. Default is the size of the interface.
MAX Retry Value (N2)	The maximum number of times the LLC transmits an RR without receiving an acknowledgment
Rcvd I-frames before ACK (N3)	Indicates the value that is used with T2 timer to reduce acknowledgment traffic for received I-frames.
Transmit window size (Tw)	Indicates the maximum number of I-frames that can be sent before receiving an RR.
Working transmit size (Ww)	The maximum number of I-frames that are sent before receiving an RR.
Acks Needed to Inc Ww (Nw)	This field is set to a default value of 1.
Current send seq (Vs)	Send state variable (Ns value for the next I-frame to be transferred).
Current Rcv seq (Vr)	Receive state variable (next in-sequence Ns to be accepted).
Last ACK'd sent frame (Va)	Acknowledged state variable (last valid Nr received).
No. of frames in ACK pend q	Number of transmitted I-frames waiting for acknowledgment.
No. of frames in transmit pend q	Number of frames waiting to be transmitted.
Local Busy	The local side of the LLC connection is sending RNRs.
Remote Busy	The remote side of the LLC is receiving RNRs.
Poll Retry count	Indicates the current value of the retry of the counter (counts down) in the LLC protocol.
Appl output flow stopped	The LLC has told the application to stop giving it outgoing data frames.
Send process running	This process runs concurrently with all other frame actions and takes I-frames in the transmit queue and sends them.
Frames Xmt and Rcvd	Displays the total number of frame types transmitted (Xmt) and (Rcvd).
I-frames discarded by LLC	Counter that displays the total number of I-frames discarded by the LLC, usually because the sequence number is out of sequence.
I-frames refused by LLC user	Counter that displays the number of I-frames discarded by the software above the LLC. For example, DLSw (Data Link Switching).

Set

Use the **set** command to dynamically configure the LLC parameters on a current LLC session. Any changes that you make to the parameters are effective for the life of session. These parameters are the same as those listed in Chapter 22, “Configuring IEEE 802.5 Token-Ring Network Interfaces” on page 22-1.

Attention: Changing LLC parameters from the default can affect how the LLC protocol works.

Syntax: set n2-max_retry *count* n3-frames-rcvd-before-ack *count* nw-acks-to-inc-ww *count* t1-reply-timer *seconds* t2-receive-ack-timer *seconds* ti-inactivity-timer *seconds* tw-transmit-window *seconds*

n2-max_retry
The maximum number of retries by LLC protocol. For example, N2 is the maximum number of times the LLC transmits an RR without receiving an acknowledgment when the inactivity timer expires. Default is 8. Minimum is 1. Maximum is 127.

Example: `set n2-max_retry`

n3-frames-rcvd-before-ack
This value is used with the T2 timer to reduce acknowledgment traffic for received I-frames. Set this counter to a specified value. Each time an I-frame is received, this value is decremented. When this counter reaches 0 or the T2 timer expires, an acknowledgment is sent. Default is 1. Minimum is 1. Maximum is 255.

Example: `set n3-frames-rcvd-before-ack`

nw-acks-to-inc-ww
This field is set to a default value of 1.

t1-reply-timer
This timer expires when the LLC fails to receive a required acknowledgment or response from the other LLC station. When this timer expires, an RR is sent with the poll bit set and T1 is started again. If the LLC receives no response after the configured maximum number of retries (N2), the link underneath is declared inoperative. Default is 1. Minimum is 1. Maximum is 256.

Example: `set t1-reply-timer`

t2-receive-ack-timer
This timer is used to delay sending of an acknowledgment for a received I-format frame. This timer is started when an I-frame is received and reset when an acknowledgment is sent. If this timer expires, LLC2 sends an acknowledgment as soon as possible. Set this value so that it is less than that of T1. This insures that the remote LLC2 peer receives the delayed acknowledgment before the T1 timer expires. Default is 1 (100 ms). Minimum is 1. Maximum is 2560.

Example: `set t2-receive-ack-timer`

Note: If this timer is set to 1 (the default) it will not run (for example, **n3-frames-rcvd-before-ack=1**).

ti-inactivity-timer
This timer expires when the LLC does not receive a frame for a specified time period. When this timer expires the LLC transmits an RR until the other LLC responds or the N2 timer expires. Default is 30 seconds. Minimum is 1 second. Maximum is 256 seconds.

Example: `set ti-inactivity-timer`

tw-transmit-window
Sets the maximum number of I-frames that can be sent before receiving an RR. Assuming that the other end of the LLC session can actually receive this many consecutive I-frames, and the router has enough heap memory to keep copies of these frames until an acknowledgment is received, increasing this

value may increase the throughput. Default is 2. Minimum is 1. Maximum is 127.

Example: `set tw-transmit-window`

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

Chapter 26. Configuring the Ethernet Network Interface

This chapter describes how to configure the Ethernet interface. It includes the following sections:

- "Accessing the Interface Configuration Process"
- "Ethernet Configuration Commands"

Accessing the Interface Configuration Process

Use the following procedure to access the configuration process. This process gives you access to an Ethernet interface's *configuration* process.

1. At the OPCON prompt (*), enter the **status** command to find the PID for CONFIG. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter **talk 6**. (For more detail on this command, refer to Chapter 2, "The OPCON Process and Commands" on page 2-1.) For example:

   ```
   * talk 6
   Config>
   ```

   The CONFIG prompt (`Config>`) displays on the console. If the prompt does not appear when you first enter configuration, press **Return** again.
3. At the CONFIG prompt, enter the **list devices** command to display the network interface numbers for which the router is currently configured. For example:

   ```
   Config> list devices

   Ifc 0 Ethernet                  CSR  81600, CSR2  80C00, vector 94
   Ifc 1 WAN X.25                  CSR  81620, CSR2  80D00, vector 93
   Ifc 2 WAN X.25                  CSR  81640, CSR2  80E00, vector 92
   Ifc 3 WAN PPP                   CSR 381620, CSR2 380D00, vector 125
   Ifc 4 WAN Frame Relay           CSR 381640, CSR2 380E00, vector 124
   Ifc 5 Token Ring                CSR 600000, vector 95
   ```

4. Record the interface numbers.
5. Enter the **network** command and the number of the Ethernet interface you want to configure. For example:

   ```
   Config> network 0
   ETH Config>
   ```

   The Ethernet configuration prompt (`ETH Config>`), is displayed.

Ethernet Configuration Commands

This section summarizes and then explains the Ethernet configuration commands. Enter the commands at the `ETH config>` prompt.

Table 26-1. Ethernet Configuration Command Summary

Command	Function
? (Help)	Displays all the Ethernet commands or lists subcommand options for specific commands.
Connector-Type	Sets the connector type.
Frame	Sets the NetWare IPX encapsulation type.
IP-Encapsulation	Sets the IP encapsulation as Ethernet (type X'0800') or IEEE (802.3 with SNAP).
List	Displays the current connector-type, NetWare IPX encapsulation, and IP encapsulation.
Physical-Address	Sets the physical MAC address.
Exit	Exits the Ethernet config process.

? (Help)

Use the **?** (**help**) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: `ETH config>` **?**

Connector-Type

Use the **connector-type** command to set the connector type. 2210s support AUI (10BASE5) and RJ-45 (10BASE-T) connectors, and auto-config options.

Syntax: connector-type *name*

Example: **`connector-type aui`**

Frame

Use the **frame** command to select the Ethernet encapsulation format used by this interface. This is required if you are using NetWare-VMS on the Ethernet, and is often used when there are ISO nodes on the same Ethernet. The following options are available:

- ethernet_II (default of NetWare 4.0 and greater) - Ethernet_II uses Ethernet version 2.0 protocol 81-37.
- ethernet_8022 - Ethernet_8022 uses Ethernet 802.3 with 802.2 SA E0.
- ethernet_8023 (default of pre-NetWare 4.0 and lower) - Ethernet_8023 uses Ethernet 802.3 without any 802.2 header.
- ethernet_SNAP - Ethernet_SNAP uses 802.3, 802.2 with SNAP PID 00-00-00-81-37.

Note: The ethernet_SNAP encapsulation is not architecturally valid and is not fast-pathed. No cache entries will appear for network entries using this encapsulation.

Syntax: frame *type*

Example: `frame ethernet_II`

IP-Encapsulation

Use the **IP-encapsulation** command to select Ethernet (Ethernet type X'0800') or IEEE 802.3 (Ethernet 802.3 with SNAP). Enter **e** or **i**.

Syntax: IP-encapsulation *type*

Example: `IP-encapsulation e`

List

Use the **list** command to display the current configuration for the Ethernet interface including the connector-type, IPX encapsulation type, and IP encapsulation type.

Syntax: list all

Example: `list all`

```
Connector type:            AUI (10BASE5)
NetWare IPX encapsulation: ETHERNET_SNAP
IP Encapsulation:          ETHER
MAC Address:               12:15:00:FA:00:FE
```

Physical-Address

Use the **physical-address** command to set the physical (MAC) address.

physical-address

This command lets you indicate whether you want to define a locally administered address for the Ethernet interface's MAC sublayer address, or use the default burned-in address (indicated by all zeros). The MAC sublayer address is the address that the Ethernet interface uses to receive and transmit frames.

Note: Pressing **Enter** leaves the value the same. Entering **0** causes the router to use the burned-in address. The default is to use the burned-in address.

Valid Values: Any 12-digit hexadecimal address.

Default Value: burned-in address (indicated by all zeros).

Example:

```
set physical-address
MAC address in 00:00:00:00:00:00 form []? 12:15:00:FA:00:FE
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

Chapter 27. Monitoring the Ethernet Network Interface

This chapter describes how to monitor the Ethernet interfaces. It includes the following sections:

Displaying Ethernet Statistics through the Interface Command

You can also use the **interface** command from the GWCON environment to display the following statistics.

```
+ interface 0
                              Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
0  0   Eth/0        81600   5E          1          1            0
Ethernet/IEEE 802.3 MAC/data-link on SCC Ethernet interface

Physical address         000093808000
RISC Microcode Revision:            1
PROM address             000093808000

Input statistics:
  failed, frame too long          0   failed, FCS error           0
  failed, alignment error         0   failed, FIFO overrun        0
  internal MAC rcv error          0   packets missed              0

Output statistics:
  deferred transmission           6   single collision            2
  multiple collisions             0   total collisions            2
  failed, excess collisions       0   failed, FIFO underrun       0
  failed, carrier sense err       0   SQE test error              0
  late collision                  0   internal MAC trans errors   0
```

These statistics have the following meaning:

Nt
: Global network number.

Nt'
: This field is for the serial interface card. Disregard the output.

Interface
: Interface name and its instance number.

CSR
: Command and status register address.

Vec
: Interrupt vector

Self-Test: Passed
: Number of self-tests that succeeded.

Self-Test: Failed
: Number of self-tests that failed.

Maintenance: Failed
Number of maintenance failures.

Physical address
The Ethernet address of the device currently in use. This may be the PROM address or an address overwritten by some other protocol.

PROM address
The permanent unique Ethernet address in the PROM for this Ethernet interface.

Interface restarts
The number of times the Ethernet chip timed out, or the Ethernet driver received a bad command from the handler. For information about why a restart occurred, refer to messages Eth.043 and Eth.044 in the *IBM Nways Event Logging System Messages Guide*

Interface type
This specifies the connector type as AUI or RJ45.

Input statistics:

failed, packet too long or failed, frame too long
The Failed, Packet Too Long counter increments when the interface receives a packet that is larger than the maximum size of 1518 bytes for an Ethernet frame. This data is exported via SNMP as the dot3StatsFrameTooLongs counter.

failed, CRC error or failed, FCS (Frame Check Sequence) error
The Failed, CRC (Cyclic Redundancy Check) Error counter increments when the interface receives a packet with a CRC error. This data is exported via SNMP as the dd3StatsFCSErrors counter.

failed, framing error or failed, alignment error
The Failed, Framing Error counter increments when the interface receives a packet whose length in bits is not a multiple of eight.

failed, FIFO over-run or failed, FIFO overrun
The Failed, FIFO (First In, First Out) Overrun counter increments when the Ethernet chipset is unable to store bytes in the local packet buffer as fast as they come off the wire.

collision in packet
The counter increments when a packet collides as the interface attempts to receive a packet, but the local packet buffer is full. This error indicates that the network has more traffic than the interface can handle.

short frame
The counter increments when the interface receives a packet with a short frame.

buffer full warnings
The Buffer Full Warnings counter increments each time the local packet buffer is full.

packets missed
The Packets Missed counter increments when the interface attempts to receive a packet, but the local packet buffer is full. This error indicates that the network has more traffic than the interface can handle.

internal mac rcv errors
Receive errors that are not late, excessive, or carrier check collisions. This data is exported via SNMP as the dot3StatsInternalMacReceiveErrors counter. This statistic is the sum of the FIFO Overruns.

Output statistics:

initially deferred or deferred transmission
The Initially Deferred counter increments when the carrier sense mechanism detects line activity causing the interface to defer transmission. This data is exported via SNMP as the dot3StatsDeferredTransmissions counter.

single collision
The Single Collision counter increments when a packet has a collision on the first transmission attempt, and then successfully sends the packet on the second transmission attempt. This data is exported via SNMP as the dot3StatsSingleCollisionFrames counter.

multiple collisions
The Multiple Collisions counter increments when a packet has multiple collisions before being successfully transmitted. This data is exported via SNMP as the dot3MultipleCollisionFrames counter.

total collisions
The Total Collisions counter increments by the number of collisions a packet incurs.

failed, excess collisions
The Failed, Excess Collisions counter increments when a packet transmission fails due to 16 successive collisions. This error indicates a high volume of network traffic or hardware problems with the network. This data is exported via SNMP as the dot3StatsExcessiveCollisions counter.

failed, FIFO underrun
The Failed, FIFO Underrun counter increments when packet transmission fails due to the inability of the interface to retrieve packets from the local packet buffer fast enough to transmit them onto the network.

failed, carrier check or failed, carrier sense error
The Failed, Carrier Check counter increments when a packet collides because carrier sense is disabled. This error indicates a problem between the interface and its Ethernet transceiver. This data is exported via SNMP as the dot3StatsCarrierSenseErrors counter.

CD heartbeat error or SQE test error
The CD (Collision Detection) Heartbeat Error or SQE (Signal Quality Error) counter increments when the interface sends a packet but detects that the transceiver has no heartbeat. The packet is treated as successfully transmitted because some transceivers do not generate heartbeats. This data is exported via SNMP as the dot3StatsSQETestErrors counter.

out of window collisions or late collisions
The Out of Window Collisions counter increments when a packet collides after transmitting at least 512 bits. This error indicates that an interface on the network failed to defer, or that the network has too many stations.

internal mac tx errors or internal MAC trans errors
Transmit errors that are not late, excessive, or carrier check collisions. This data is exported via SNMP as the dot3StatsInternalMacTransmitErrors counter. This statistic is the sum of the FIFO Underruns.

RISC Microcode Version:
This gives the version of the microcode running in the RISC controller of the communications processor module.

Accessing the Interface Console Process

To monitor information related to the Ethernet Network Interface, access the interface console process by doing the following:

1. Enter the **status** command to find the PID for GWCON. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter **talk 5**. For example:

   ```
   * talk 5
   ```

   The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter GWCON, press **Return** again.
3. At the GWCON prompt, enter the **configuration** command to see the protocols and networks for which the router is configured. For example:

   ```
   + configuration
   ```

   See page “Configuration” on page 6-6 for sample output of the **configuration** command.
4. Enter the **network** command and the number of the Ethernet interface. In this example:

   ```
   + network 0
   ETH>
   ```

   The Ethernet console prompt is displayed. You can now view information about the Ethernet interface by entering console commands.

Ethernet Interface Console Commands

This section summarizes and explains the Ethernet console commands. Enter commands at the ETH> prompt. Table 27-1 lists the console commands.

Table 27-1. Ethernet Console Command Summary

Command	Function
? (Help)	Displays all the Ethernet commands or lists subcommand options for specific commands.
Collisions	Displays collision statistics for the specified Ethernet interface.
Exit	Exits the Ethernet monitoring process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a *?* after a specific command name to list its options.

Syntax: ?

Example:

```
Eth> ?
collisions
exit
```

Collisions

This command shows the counts of transmissions for packets that incurred collisions before successful transmission. Counters are given for packets sent after the collision XXXXx packets sent after 15 collisions. Increasing numbers of packets transmitting with collisions and higher numbers of collision per packet are signs of transmitting onto a busy Ethernet.

These counters are cleared by the OPCON **CLEAR** command. This data is exported via SNMP as the dot3CollTable counter.

Syntax: collisions

Example: Eth> **coll**

```
Transmitted with  1 collisions:0
Transmitted with  2 collisions:0
Transmitted with  3 collisions:0
Transmitted with  4 collisions:0
Transmitted with  5 collisions:0
Transmitted with  6 collisions:0
Transmitted with  7 collisions:0
Transmitted with  8 collisions:0
Transmitted with  9 collisions:0
Transmitted with 10 collisions:0
Transmitted with 11 collisions:0
Transmitted with 12 collisions:0
Transmitted with 13 collisions:0
Transmitted with 14 collisions:0
Transmitted with 15 collisions:0
```

Exit

Use the **exit** command to return to the previous prompt level (GWCON).

Syntax: exit

Example: Eth> **exit**

Chapter 28. Configuring Serial Line Interfaces

This chapter describes the interface configuration process for a serial interface and includes the following sections:

- "Accessing the Interface Configuration Process"
- "Network Interfaces and the GWCON Interface Command"

IMPORTANT: To configure Frame Relay, PPP, X.25, V.25bis, SDLC Relay, and SDLC protocols on the serial interface, use the commands in this chapter and then refer to the commands in the chapters that describe the specific protocol.

See "Configuring the Network Interface" on page 1-19 for a table of protocols and the interfaces that support those protocols.

Accessing the Interface Configuration Process

To access the interface configuration process for a serial interface, first access the `Config>` prompt and issue the command **set data-link**. Next, at the `Config>` prompt, enter the interface type and number to access the configuration environment for the interface.

For example, to configure a serial interface for X.25, you must access the `X.25 config>` environment by issuing the following commands:

```
Config> set data-link X25 2
Config> network 2
```

From the `X.25 config>` environment, you can complete your configuration of X.25 on the serial interface. See Chapter 29, "Configuring the X.25 Network Interface" on page 29-1.

When you are done configuring the serial interface, enter the **restart** command after the OPCON prompt (*) and respond **yes** to the prompt to enable the new configuration.

Network Interfaces and the GWCON Interface Command

While serial line interfaces do not have their own console process for monitoring purposes, routers can display complete statistics for all installed network interfaces when you use the **interface** command from the GWCON environment. For more information on the **interface** command and displaying statistics, see Chapter 6, The GWCON (Monitoring) Process and Commands.

Chapter 29. Configuring the X.25 Network Interface

The X.25 network interface connects a router to an X.25 virtual circuit switched network. The X.25 network interface software and hardware allows the router to communicate over a public X.25 network. The X.25 network interface complies with CCITT 1980, CCITT 1984, CCITT 1988 and ISO 8208 1990 specifications for X.25 interfaces offering multiplexed channels and reliable end-to-end data transfer across a wide area network.

This chapter includes the following sections:

For information on configuring X.25 Transport Protocol (XTP) for transporting X.25 traffic over TCP/IP, refer to *Protocol Configuration and Monitoring Reference Volume 2 for Nways Multiprotocol Routing Services Version 2.1*.

Basic Configuration Procedures

This section outlines the minimal configuration steps required to get the X.25 interface up and running. The X.25 parameters must be consistent with the X.25 network the interface on the router will connect to. For more information, refer to the configuration commands described in this chapter.

Note: You must restart the router for the configuration changes to take effect.

1. At the OPCON prompt (*), type **talk 6**.

   The `Config>` prompt appears.

2. Type **list devices** to display a list of the interfaces from which you can select. Use the appropriate interface number in the following step.

3. Type **set data-link x25**.

   The `Interface Number [0]?` prompt appears.

4. Type the appropriate interface number.

5. Connect to the network by typing **net** *#* at the `Config>` prompt.

   The `X.25 Config [#]>` prompt appears.

6. At this prompt, type **set address** *x.25-node-address*.

   The X.25 address is a unique X.121 address that is used during call establishment. You ***must*** specify an X.25 node address for each router interface. Use the **add htf-addr** and **set htf-addr** commands to configure the x.121 addresses. Failure to set the network address prevents the X.25 interface from joining the attached network.

7. Type **set equipment-type** and specify whether the frame and packet levels act as DCE or DTE. The default for this command is DTE.

8. Type **set svc** and define the lowest and highest SVCs that you are using. The default is for 1 SVC.

9. Type **add protocol** *protocol_name* to add the protocols that will be running over the X.25 interface. You will be prompted for window size, default packet size, maximum packet size, circuit idle time, and max VCs.

   Note: You need to add the protocols only once for all X.25 networks on the router.

10. Type **add address** *protocol_name* to add an address translation for each protocol's destination address reachable over this interface.
11. Type **exit** to return to the `Config>` prompt.
12. Press **Ctrl** **P** to return to the OPCON prompt (*).
13. Type **restart** and respond **yes** to the prompt.

Setting the National Personality

Each public data network, such as GTE's Telenet or DDN's Defense Data Network, has its own standard configuration. The term *National Personality* specifies a group of variables used to define a public data network's characteristics. The configuration information in the National Personality provides the router with control information for packets being transferred over the link. The National Personality option defines 27 default parameters for each public data network.

To view the configuration values that are in your X.25 National Personality, execute the X.25 configuration **list detailed** command. Configure each public data network connected to the router by executing the X.25 configuration **national-personality set** command.

The National Personality is a generalized template for network configuration. If necessary, you can individually configure each frame and packet layer parameter.

Understanding the X.25 Defaults

The following tables list the defaults for the various parameters for the X.25 *set, national set* and *national enable* commands.

Table 29-1 (Page 1 of 2). Set Command

Parameter	Default
address ...	none
cable	none
calls-out ...	4
clocking ...	external
default-window-size ...	2
encoding	NRZ
equipment-type ...	DTE
htf addr ...	none
inter-frame-delay ...	0
mtu	1500
national-personality ...	GTE Telenet
pvc ...	low=0 high=0
speed	9600

Table 29-1 (Page 2 of 2). Set Command

Parameter	Default
svc	low inbound=0, high inbound=0 low 2-way=1, high 2-way=64 low outbound=0, high outbound=0
throughput-class ...	inbound=outbound=2400
vc-idle ...	30

Table 29-2. National Enable Parameters

Parameter	DDN Default	GTE Default
accept-reverse-charges	off	on
flow-control-negotiation	on	on
frame-ext-seq-mode	off	off
packet-ext-seq-mode	off	off
request-reverse-charges	off	on
suppress-calling-addresses	off	off
throughput-class-negotiation	on	on
truncate-called-addresses	off	off

Table 29-3. National Set Parameters

Paramter	DDN Default	GTE Default
call-req	20 decaseconds	20 decaseconds
clear-req ...	retries=1	retries=1
	18 decaseconds	18 decaseconds
disconnect-procedure ...	passive	passive
dp-timer	500 milliseconds	500 milliseconds
frame-window-size	7	7
n2-timeouts	20	20
packet-size ...	128, max=256	128, max=256
reset ...	retries=1	retries=1
	18 decaseconds	18 decaseconds
restart ...	retries=1	retries=1
	18 decaseconds	18 decaseconds
min-recall	10 seconds	10 seconds
min-connect	90 seconds	90 seconds
collision-timer	10 seconds	10 seconds
standard-version	1984	1984
t1-timer	4 seconds	4 seconds
t2-timer	0	0
truncate-called-addr-size	2	2

X.25 Configuration Commands

This section summarizes and explains all the X.25 configuration commands.

The X.25 configuration commands allow you to specify network parameters for router interfaces that transmit X.25 packets. The information you specify with the configuration commands activates when you restart the router.

Enter the X.25 configuration commands at the `X.25 config>` prompt. Table 29-4 shows the commands.

Table 29-4. X.25 Configuration Commands Summary

Command	Function
? (Help)	Lists the interface configuration commands or lists the options associated with specific commands.
Set	Sets the local and DDN X.25 node addresses, window size for packet levels, identifies the National personality, the MTU, and the maximum number of calls. Defines the PVC and SVC channel ranges, the number of seconds that a switched circuit can be idle before it is cleared, and specifies whether one router needs to act as a DCE (when two routers are directly connected without an intervening X.25 network) or the more normal method of acting at a DTE connected to an X.25 network. Sets speed, encoding, clocking, throughput class, and cable type.
Enable/Disable	Enables/Disables incoming-calls-barred feature, outgoing-calls-barred feature, dynamic DDN address translations, and lower-dtr feature.
National Enable or National Disable	Enables/Disables the parameters defined by the National Personality configuration.
National Set	Sets parameters defined by the National Personality configuration.
National Restore	Restores the National Personality configuration to its default values.
Add/Change/Delete	Adds/Changes/Deletes an address translation, a protocol encapsulation, or a PVC definition.
List	Lists the defined address translations, National Personality parameters, protocol encapsulation, or PVC definitions.
Exit	Exits the X.25 configuration process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
Add
Change
Delete
Disable
Enable
List
National-Personality
Set
Exit
```

Set

Use the **set** command to configure local X.25 node addresses, maximum number of calls, frame and packet level window size, lowest to highest PVC and SVC channels, and the idle time for a switched circuit.

Syntax: set
- address . . .
- cable
- calls-out . . .
- clocking . . .
- default-window-size . . .
- encoding
- equipment-type . . .
- htf addr . . .
- inter-frame-delay . . .
- mtu
- national-personality . . .
- pvc . . .
- speed . . .
- svc
- throughput-class . . .
- vc-idle . . .

address *x.25-node-addr*

Sets the local X.25 interface address (*x.25-node-addr*). Set the X.25 node address to 0, not to 00, to delete the local X.25 address.

Example: `set address 8982800`

cable *type*

Sets the cable type as follows:

- RS-232 DTE
- RS-232 DCE
- V35 DTE
- V35 DCE
- V36 DTE
- X21 DTE
- X21 DCE

A DTE cable is used when you are attaching the router to some type of DCE device (for example, a modem or a DSU/CSU).

A DCE cable is used when the router is acting as the DCE and providing the clocking for direct attachment.

Example: `set cable RS-232 DTE`

calls-out *value*

Sets the maximum number of locally initiated, simultaneously active SVCs.

Valid Values: 1 to 239

Default Value: 4

Example: `set calls-out 3`

clocking *external* or *internal*

To connect to a modem or DSU, configure clocking as external. To connect directly to another DTE device, use a DCE cable, set the clocking to internal, and configure the line speed.

external

Example: `set clocking internal`

default-window-size *svalue*

Sets the window size for the packet level assigned by the router if there is no window-size facility in the Call-Request packet. The range is determined by the National Personality packet modulus (PACKET-EXT-SEQ-MODE).

Default: 2

Example: `set default-window-size 3`

encoding *NRZ* OR *NRZI*

Sets the HDLC transmission encoding scheme for the interface. Encoding may be set for NRZ (non-return to zero) or NRZI (non-return to zero inverted). NRZ is the more widely used encoding scheme while NRZI is used in some IBM configurations.

Default: NRZ

Example: `set encoding nrz`

equipment-type *DCE* OR *DTE*

Specifies whether the frame and packet levels act as DCE or DTE. This command has no relation to the cable type in use.

Default: DTE

Example: `set equipment-type DCE`

htf addr *x.25-node-addr*

Sets the local DTE address when DDN is used. It converts the IP address to an X.121 address as opposed to the **set address** command, which is used to set the local DTE address when CCITT is used.

Example: `set htf-address 11.42.0.137`

inter-frame-delay *value*

This parameter defines the minimum delay between transmitted frames. Setting this parameter is useful when interfacing directly to older equipment which may not be able to consistently handle consecutive frames separated by one flag (resulting in receive errors such as T1 timeouts).

The IBM 2210 requests from 0 to 15 extra flags between frames.

Default: 0

Example: `set inter-frame-delay 1`

mtu *value*

Sets the Maximum Transmit Unit (MTU) in bytes. This is the maximum message size that will be delivered to the X.25 interface to package and transmit over the serial line. The range is 576 to 16384 .

Default: 1500

If you are encountering packet reassembly timeouts when transferring data over the X.25 interface, you should determine what the minimum packet size is for all LAN or serial interfaces that lead to the end-point, then calculate a more suitable X.25 MTU. You should not directly consider the actual X.25 packet size in this calculation because X.25 tends to use a smaller packet size. X.25 usually sends up to 7 packets at one time before waiting for an acknowledgment.

For example, consider a network topology that includes:

- A Token-Ring LAN having a packet size of 4000
- An X.25 serial line having a packet size of 128 with a window size of 7 and a bit rate of 9600 bps
- An Ethernet LAN with a packet size of 1500

In this case, you should probably set the X.25 MTU to 1500. That means that about 12 packets will be sent over the X.25 interface. (MTU / X.25 packet size = number of X.25 packets to be sent).

When using an MTU of 4096, 32 packets must be sent over the X.25 interface. (4000 /128 = 31.25). In this case, packet reassembly timeouts will probably occur if the X.25 modem speed is 9600 bps. Using an X.25 modem speed of 56 Kbps would probably solve this problem.

Note: The MTU parameter has significant impact on the memory requirements and memory utilization of the device. Use an MTU value of 8192 or less for devices with less than 8M of memory. Table 29-5 shows the recommended maximum number of virtual circuits based on a device with one X.25 interface.

Table 29-5. Virtual Circuit Limits based on Memory and MTU Size

Device Memory	MTU Size	Number of Virtual Circuits
4 MB	8 KB	239
	16 KB	Not recommended
8 MB	8 KB	239
	16 KB	?
>16 MB	8 KB	239
	16 KB	239

Example: `set mtu 2048`

national-personality *GTE-Telenet* or *DDN*
Sets the 28 default parameters for either GTE-Telenet or DDN National Personality.

Default: GTE-Telenet

Example: `set national-personality DDN`

pvc low/high *value*
Defines the lowest to the highest Permanent Virtual Circuit channel number. Zero indicates no PVCs. By default there are no PVCs.

pvc low 0

pvc high 0

The range is 1 to 4095. These values are setting boundaries of a given VC range. There is a maximum of 239 VCs in the 2210. There is a maximum of 200 PVCs.

Example: `set pvc low 40`

Note: Values must not overlap values set for SVCs.

speed *speed-setting*

For internal clocking, this command specifies the speed of the transmit and receive clock lines.

Valid values: 2400 to 2048000 bps.

For external clocking, this command does not affect the hardware but it sets the speed some protocols, such as IPX, use to determine routing cost parameters. In these cases, set the speed to match the actual line speed. If the speed is not configured, the protocols assume a speed of 1000 000 bps when calculating routing cost parameters. The maximum line speed that can be configured if external clocking is 6 312 000 bps.

Default: 9600

Note: The X.25 software is supported only at speeds up to 256 000 bps.

Example: `set speed 19200`

svc low/high *inbound* OR *two-way* OR *outbound value*

Defines the lowest to the highest switched virtual circuit channel number. When low=high=0, no VCs in this category are defined.

Note: The total number of inbound, outbound, and two-way SVCs cannot exceed 239.

Example: `set SVC low-two-way 1`

Inbound Specifies the range of logical channel numbers to be assigned to inbound SVCs. By default, there are no inbound-only SVCs.

Valid values: 0 to 4095

Default values: 0

Two-way Specifies the range of logical channel numbers to be assigned to two–way SVCs. By default, there are sixty-four 2-way SVCs.

Valid values: 0 to 4095

Default values:

svc low 1

svc high 64

Outbound Specifies the range of logical channel numbers to be assigned to outbound SVCs. By default, there are no outbound-only SVCs.

Valid values: 0-4095

Default: 0

Note: Values in each range must not overlap other SVC ranges nor the PVC range. Table 29-6 on page 29-9 shows a possible VC configuration.

Table 29-6. Example VC Definitions. Only 239 VCs can be used from this definition

	Low	High
PVC	1	40
inbound	0	0
two-way	41	59
outbound	60	500

throughput-class inbound ***or*** outbound *bit-rate*

Defines the throughput class requested when making a call request while throughput negotiation is enabled.

Default: 2400 bps

This setting is ignored when processing incoming `call requests`.

Example: **`set throughput-class inbound`**

vc-idle *value*

Defines the number of seconds that a switched circuit can be idle before it is cleared by the router. Zero indicates that the router never clears an idle circuit.

Valid values: 1 to 255

Default: 30 seconds

Example: **`set vc-idle 40`**

Enable

Use the **enable** command to enable DDN address translations, interface resets, or the incoming-calls-barred, outgoing-calls-barred, and lower-dtr features.

Syntax: enable ddn—address-translations

Note: Enabling ddn-address-translations is no longer allowed. This feature defaults to enabled when the national personality selected is DDN, and defaults to disabled in all other cases.

incoming-calls-barred
lower-dtr
outgoing-calls-barred

incoming-calls-barred

Specifies that the router will not accept incoming calls. The default setting for this parameter is disabled or *off*, which allows incoming calls.

Example: **`enable incoming-calls-barred`**

lower-dtr

This parameter determines the way the data terminal ready (DTR) signal is handled for leased serial-line interfaces that are disabled. If this parameter is set to "disabled" (the default), the DTR signal will be raised when the interface is disabled.

If *lower-dtr* is set to "enabled," the DTR will be lowered when the interface is disabled. This behavior may be desirable in situations where the interface has been configured as an alternate link for WAN Reroute and the interface is

connected to a dial-out modem which maintains its dial connection based on the state of the DTR signal.

When lower-dtr is enabled and the interface is disabled, the DTR signal is low and the modem keeps the dial connection down. When the interface is enabled, due to a WAN Reroute backup scenario, DTR is raised and the modem dials a stored number to the backup site. When the primary interface is restored, the alternate interface is disabled, DTR is lowered, and the modem hangs up the dial connection.

The following cable types are supported:

RS-232
V.35
V.36

The default setting is disabled.

Example: `enable lower-dtr`

outgoing-calls-barred

Specifies that the router will not allow outgoing calls. The default setting for this parameter is disabled or *off*, which allows outgoing calls.

Example: `enable outgoing-calls-barred`

Disable

Use the **disable** command to disable DDN address translations, interface resets as part of network certification, or the incoming-calls-barred or outgoing-calls-barred features.

Note: If you set DDN as the national personality, DDN address translation is enabled automatically and this parameter has no effect.

Syntax: disable ddn-address-translations

Note: Disabling ddn-address-translations is no longer allowed. This feature defaults to enabled when the national personality selected is DDN, and defaults to disabled in all other cases.

incoming-calls-barred
lower-dtr
outgoing-calls-barred

National Enable

Use the **national enable** command to enable a feature defined in the National Personality configuration.

Syntax: national enable

accept-reverse-charges
flow-control-negotiation
frame-ext-seq-mode
packet-ext-seq-mode
request-reverse-charges
suppress-calling-addresses
throughput-class-negotiation
truncate-called-addresses

accept-reverse-charges
Accepts reverse charge calls during call establishment. This option is not available for DDN.

DDN Default off

GTE Default on

Example: `national enable accept-reverse-charges`

flow-control-negotiation
Enables the negotiation of packet and window size during call setup of SVCs.

DDN Default on

GTE Default on

Example: `national enable flow-control-negotiation`

frame-ext-seq-mode
Sets the frame layer sequence numbering to modulo 128 (i.e., 0 through 127).

DDN Default off

GTE Default off

Example: `national enable frame-ext-seq-mode`

packet-ext-seq-mode
Enables the packet layer to use extended sequence numbers (0 through 127).

DDN Default off

GTE Default off

Example: `national enable packet-ext-seq-mode`

request-reverse-charges
Requests reverse charges for all outgoing calls.

DDN Default off

GTE Default on

Example: `national enable request-reverse-charges`

suppress-calling-address
Suppresses the source address in call packets.

DDN Default off

GTE Default off

Example: `national enable suppress-calling-addresses`

throughput-class-negotiation
Enables the registration of throughput class.

DDN Default off

GTE Default on

Example: `national enable throughput-class-negotiation`

truncate-called-addresses
Enables truncation of the called DTE address when transmitting a call to a DTE. This option applies only to XTP circuits.

DDN Default off

GTE Default off

Example: `national enable truncate-called-addresses`

National Disable

Use the **national disable** command to disable a feature defined by the National Personality configuration.

Syntax: national disable
accept-reverse-charges
flow-control-negotiation
frame-ext-seq-mode
packet-ext-seq-mode
request-reverse-charges
suppress-calling-addresses
throughput-class-negotiation
truncate-called-addresses

National Set

Use the **national set** command to set one or all of the default values made to the National Personality configuration.

Syntax: national set
call-req
clear-req . . .
disconnect-procedure . . .
dp-timer
frame-window-size
n2-timeouts
packet-size . . .
reset . . .
restart . . .
min-recall
min-connect
collision-timer
standard-version
t1-timer
t2-timer
truncate-called-addr-size

call-req

Specifies the number of 10-second intervals permitted before giving up on a call request and clearing it. A zero indicates an infinite wait. In a list command output, this is displayed as the t21 timer.

DDN Default 20 decaseconds

GTE Default 20 decaseconds

Example: `national set call-req 20`

clear-req *retries* OR *timer*

Specifies the number of clear request retransmissions.

Retries Number of clear request transmissions permitted before action is taken. In a list command output, this is displayed as the r23 retry count.

DDN Default retries=1

GTE Default retries=1

Timer Number of 10–second intervals to wait before retransmitting a clear request packet. A zero in the timer value indicates an indefinite wait. In a list command output, this is displayed as the t23 timer.

DDN Default 18 decaseconds

GTE Default 18 decaseconds

Example: `national set clear-req retries 2`

disconnect-procedure *passive* OR *active*
Specifies the type of disconnect procedure to use when disconnecting.

DDN Default passive

GTE Default passive

Example: `national set disconnect-procedure active`

Passive Specifies that DISC frames are not used when disconnecting.

Active Specifies that DISC frames are used when disconnecting.

dp-timer
Specifies the number of milliseconds that the frame level remains in a disconnected state. Zero indicates immediate transition from disconnected phase to link setup state.

DDN Default 500 milliseconds

GTE Default 500 milliseconds

Example: `national set dp-timer 300`

frame-window-size
Specifies the number of frames that can be outstanding before acknowledgment.

DDN Default 7

GTE Default 7

Example: `national set frame-window-size 7`

n2-timeouts
Specifies the number of times the retransmit timer (T1) can expire before the interface is recycled.

DDN Default 20

GTE Default 20

Example: `national set n2-timeouts 10`

packet-size *default* OR *maximum* OR *window*
Specifies the size of the packet.

Example: `national set packet-size maximum 256`

default Number of bytes in the data portion of the packet. Possible options include 128, 256, 512, 1024, 2048, and 4096. This value is used in the absence of packet size negotiation. *Default* cannot be greater than *maximum*.

DDN Default 128

GTE Default 128

maximum Maximum number of bytes in the data portion of the packet. Possible options include 128, 256, 512, 1024, 2048, and 4096.

DDN Default 256

GTE Default 256

window Number of outstanding I-frames permitted before acknowledgment is required. The range is determined by the National Personality Packet Modulus.

Related configuration parameters are

- Protocol max default window
- Set default window size

reset *retries* OR *timer*

Specifies the number of reset request retransmissions.

Example: `national set reset retries 2`

retries Number of reset request transmissions permitted before the call is cleared. The range is 0 to 255. In a list command output, this is displayed as the r22 retry count.

DDN Default 1

GTE Default 1

timer Number of 10–second intervals to wait before retransmitting a reset request packet. The range is 0 to 255. A zero in the timer value indicates an indefinite wait. In a list command output, this is displayed as the t22 timer.

DDN Default 18 decaseconds

GTE Default 18 decaseconds

restart *retries* OR *timer*

Specifies the number of restart request transmissions.

Example: `national set restart timer 12`

retries Number of restart request transmissions permitted before the interface is recycled. The range is 0 to 255. In a list command output, this is displayed as the r20 retry count.

DDN Default 1

GTE Default 1

timer Number of 10–second intervals to wait before retransmitting a restart request packet. The range is 0 to 255. A zero in the timer value indicates an indefinite wait. In a list command output, this is displayed as the t20 timer.

DDN Default 18 decaseconds

GTE Default 18 decaseconds

min-recall

Specifies the minimum number of seconds to wait prior to reinitiating a call to open an SVC. The range is 0 to 255 seconds.

DDN Default 10 seconds

GTE Default 10 seconds

Example: `national set min-recall`

min-connect
Specifies in seconds, the minimum amount a time an SVC will remain established once the connection is made barring any error conditions. The range is 0 to 255 seconds.

DDN Default 90 seconds

GTE Default 90 seconds

Example: `national set min-connect`

collision-timer
Specifies in seconds, the time delay used prior to reinitiating a call to open an SVC if the original attempt resulted in a call collision. The range is 0 to 255 seconds.

DDN Default 10 seconds

GTE Default 10 seconds

Example: `national set collision-timer`

standard-version
Options are none, v1980, v1984, and v1988.

DDN Default 1984

GTE Default 1984

Example: `national set standard-version v1984`

| t1-timer
Specifies the frame retransmit time in seconds. The range is 1 to 255.

DDN Default 4 seconds

GTE Default 4 seconds

Example: `national set t1-timer 3.2`

| t2-timer
| Specifies the amount of time in seconds to delay before acknowledging an I-frame. This is an optimization parameter. Setting the timer to 0 disables it. The range is 0 to 255.

| **DDN Default** 0

| **GTE Default** 0

| **Example:** `national set t2-timer 2`

| truncate-called-addr-size
| Specifies the number of characters truncated from the end of a called address. This parameter pertains only to XTP circuits. The range is 0 to 10.

| **DDN Default** 2

| **GTE Default** 2

| **Example:** `national set truncate-called-addr-size 4`

National Restore

Use the **national restore** command to restore one or all of the default values made to the National Personality configuration via the **national set**, **national enable**, or **national disable** command.

Syntax: national restore

- all
- accept-reverse-charges
- call-req
- clear-req . . .
- disconnect-procedure . . .
- dp-timer
- flow-control-negotiation
- frame-ext-seq-mode
- frame-window-size
- min-collission-timer
- min-connect-timer
- min-recall-timer
- network-type . . .
- n2-timeouts
- packet-size . . .
- packet-ext-seq-mode
- request-reverse-charges
- reset . . .
- restart . . .
- standard-version
- suppress-calling-addresses
- throughput-class-negotiation
- t1-timer
- t2-timer
- truncate-called-addresses
- truncate-called-addr–size

Add

Use the **add** command to add an X.121 address, a DDN X.25 Address, a protocol configuration, or a PVC definition.

Syntax: add

- address
- htf-address
- protocol
- pvc

address

Adds an X.121 address translation for a protocol supported in the configuration of the router. The prompts that appear depend on the protocol address that you are adding. (See the following examples.) The protocol address and X.121 address being entered represent the protocol and X.121 DTE address of the remote DTE connecting to the router X.25 interface. The **set address** command is used to set the local X.25 address.

Example: `add address`

IP example:

```
Protocol [IP]?  IP
IP Address [0.0.0.0]? 128.185.1.2
Enc Priority 1 []?
Enc Priority 2 []?
Enc Priority 3 []?
X.25 Address []? 1234590
```

IPX example:

```
Protocol [IP]?  IPX
CUD Field Usage (Standard or Proprietary)
IPX Host Number (in hex) []?
Enc Priority 1 []?
Enc Priority 2 []?
X.25 Address []?
```

Protocol Specifies the protocol type of the address mapping you are adding. The valid values are APPN, DECnet, DLSw, IP, IPX and VINES. The default is IP.

Enc Priority
Determines the encapsulation type, as defined in RFC 1356, that will be put in the CUD. For IP, valid choices are CC, or SNAP. For IPX, valid choice is SNAP. Enc Priority 1 is used in the first call attempt; if this fails, then Priority 2 is used and so on.

IP Address
Specifies the destination's IP address.

CUD Field Usage
This field is for IPX to X.25 address mapping only. It determines how the Call User Data (CUD) field is filled in when call request packets are received for IPX. The CUD field can be either Standard or Proprietary. Standard indicates that the usage is protocol multiplexing used in RFC 1356. Proprietary indicates a proprietary CUD field that can only be used with 2210 or compatible routers. The default is Standard.

IPX Host Number
Specifies the IPX host number of the destination.

X.25 Address
Specifies the X.121 DTE address of the remote DTE connecting to the router X.25 interface. The maximum address length is 15 digits.

htf-address
Adds a Defense Data Network (DDN) X.25 address translation.

Example: **add htf-address**

```
Protocol [IP]
Current HTF address
```

Protocol Specifies the protocol that you are running over the X.25 interface. DDN supports IP only.

Current HTF address
Specifies the destination X.121 address in Host Table Format (HTF) format. Also see ddn-address-translations in the Enable/Disable commands section.

protocol
Enables a protocol encapsulation and defines the associated parameters.

Example: **add protocol**

```
Protocol [IP]
Window Size [2]
Default Packet Size [128]
Maximum Packet Size [256]
Circuit Idle Time [30]
Maximum VCs [4]
```

QLLC example:

```
X.25 Config> add prot
Protocol [IP]? dls
Circuit Idle time [20]?
QLLC response timer [20]?
QLLC response count [10]?
Accept Reverse Charges [N]?
Request Reverse Charges [N]?
Station Type (1) PRI (2) SEC (3) PEER [3]?
Window size [128]?
Max message size [256]?
Call User Data (in HEX) [0000000000000000]?
```

Protocol Specifies which protocol's encapsulation parameters you want to add: APPN, XTP, IP, DECnet, IPX, DLSw, or Banyan VINES. The default is IP.

Window Size Specifies the maximum negotiable packet window size, the number of packets that can be outstanding before requiring packet confirmation. The default is 2. The window size can be negotiated down to 1 by the called DTE.

Related configuration parameters are:

- Set Default Window

Default Packet Size

Specifies the default requested packet size for SVCs. This value serves as the lowest negotiable packet size and must be equal to or less than the maximum packet size specified with the **national set packet-size** command. The maximum *default packet size* is 4096 bytes. The default value for this parameter is 128 bytes.

Related configuration parameters are:

- National Set Packet Size Default
- National Set Packet Size Maximum

Maximum Packet Size

Specifies the maximum negotiable packet size for SVCs. This value must be equal to or less than the maximum packet size specified with the **national set packet-size** command. The default value for this parameter is 256 bytes. The maximum value that can be configured for this parameter is 4096 bytes. This value is utilized in calculating the maximum frame size for this X.25 interface.

Related configuration parameters are:

- National Set Packet Size Default
- National Set Packet Size Maximum

Circuit Idle Time

Specifies the number of seconds that an SVC can be idle before it is cleared by the router. The range is 0 to 65365. The

default is 30 seconds. A 0 (zero) specifies that the circuit is never cleared by the router.

Maximum VCs Specifies the maximum number of circuits that are open to the same DTE address for a protocol. Refer to RFC 1356 for information on utilizing this parameter. The Valid range is 1 to 10. The default is 4.

The following are QLLC unique parameters:

QLLC response timer
The number of seconds to wait for a Q-response packet before retransmitting.

QLLC response count
The maximum number of times QLLC will retransmit. Upon exhausting this number of retries, the upper layer is notified which may result in the circuit being cleared or reset by the router.

Accept Reverse Charges
Allows this protocol to override the setting of this National Personality parameter. This does not affect the National Personality parameter.

Request Reverse Charges
Allows this protocol to override the setting of this National Personality parameter. This does not affect the National Personality parameter.

Station Type Specifies the default station type for this protocol:

Pri Primary Station
Sec Secondary Station
Peer Peer Station

Max message size
The maximum message size for this protocol. Specify a value that is less than, or equal to, the Max MTU size of the interface.

Call User Data Specifies the default CUD field used in call packets for this protocol. Specify from 1-to-16 characters. If you do not specify characters, the default 0xC3 is used.

pvc

Adds PVC, window size, and packet size definitions.

Example: **`add pvc`**

IP example:

```
Protocol [IP]?  IP
Packet Channel [1]?
Destination X.25 Address[]?
Window Size [2]?
Packet Size [128]?
```

Protocol Specifies which protocol's encapsulation you want to modify: APPN, XTP, DECnet, Banyan Vines, DLSw, IP or IPX. The default is IP.

Packet Channel
Specifies the circuit number of the PVC.

Destination X.25 Address
Specifies the X.25 address of the PVC's destination.

Window Size Specifies the number of packets that can be outstanding before requiring packet confirmation. The default is 2.

Related configuration parameters are:

- Set Default Window

Packet Size Specifies the maximum negotiable packet size for PVCs. This value must be equal to or less than the maximum packet size specified with the **national set packet-size** command. The default value for this parameter is 128 bytes. The maximum value that may be configured for this parameter is 4096 bytes. This value is utilized in calculating the maximum frame size for this X.25 interface.

Related configuration parameters are:

- Nat Set Packet Size Default
- Nat Set Packet Size Maximum

Change

Use the **change** command to change an X.121 address, an DDN X.25 Address, a protocol configuration, or a PVC definition.

Note: To change an IP address that is associated with an X.121 address, you must delete the record that contains the address correlation, then redefine the address mapping.

Syntax: change address
htf-address
protocol
pvc

address
Modifies a X.121 address translation. The prompts that appear depend on the protocol that is changing.

Example: **`change address`**

IP example:

```
Protocol [IP]  IP
IP Address [0.0.0.0]?
Enc Priority []?
X.25 Address [00000124040000]?
```

IPX example:

```
Protocol [IP]  IPX
CUD Field Usage (Standard or Proprietary) [Standard]?
IPX Host number (in hex) []?
Enc Priority []?
X.25 Address [00000124040000]?
```

htf address
Changes a Defense Data Network (DDN) X.25 address translation.

Example: **`change htf-address`**

```
Protocol [IP]
Change HTF address [0.0.0.0]?
New HTF address [10.4.0.124]?
```

protocol

Changes a protocol configuration definition.

Example: **`change protocol`**

```
Protocol [IP]
Window Size [2]
Default Packet Size [128]
Maximum Packet Size [256]
Circuit Idle Time [30]
Maximum VCs [6]
```

QLLC example:

```
X.25 Config> change prot
Protocol [IP]? dls
Idle Timer [30]?
QLLC response timer (in decaseconds) [15]?
QLLC response count [255]?
Accept Reverse Charges [N]?
Request Reverse Charges [N]?
Station Type (1) PRI (2) SEC (3) PEER [3]?
Max Packet Size [256]?
Packet Window size [7]?
Max message size [2048]?
Call User Data (in HEX, 0 for Null) []? C3010000525450
```

pvc

Changes PVC, window size, and packet size definitions.

Note: To change the protocol, packet channel or destination X.25 address, you must delete the record which contains the definition, then add it back with the changed parameters.

Example: **`change pvc`**

IP example:

```
Protocol [IP]?  IP
Packet Channel [1]?
Destination X.25 Address []?
Window Size [2]?
Packet Size [128]?
```

Delete

Use the **delete** command to delete an X.121 address, a protocol configuration definition, or a PVC definition.

Syntax: delete address
protocol . . .
pvc

address

Deletes an X.121 address translation.

Example: **`delete address`**

IP example:

```
Protocol [IP]?
IP Address [0.0.0.0]?
```

IPX example:

```
Protocol [IP]? IPX
IPX Host Number (in hex) [2]?
```

protocol *prot-type*

Deletes a protocol encapsulation configuration definition. *Prot-type* is the name or number of the protocol encapsulation that is currently defined in the router's configuration.

Example: **delete protocol IPX**

pvc

Deletes a PVC definition.

Example: **delete pvc**

```
Protocol [IP]?
Destination X.25 Address []?
```

List

Use the **list** command to display the current configuration for the specified parameter.

Syntax: list address
all
detailed
protocols
pvc
summary

address

Lists all the X.121 address translations.

Example: **list address**

```
IF#      Prot #     Active Enc     Protocol ->    X.25 address
1         0(IP)        CC          10.1.2.3 ->    1238765742
1         7(IPX)       SNAP        10       ->    12389
```

all

Lists all the X.25 addresses, National Personality parameters, all defined protocols and their values, and all defined PVCs.

Example: **list all**

```
X.25 Configuration Summary

Node Address:       313131
Max Calls Out:           4
Inter-Frame Delay:       0    Encoding:  NRZ
Speed:           64000        Clocking:  Internal
MTU:              2048        Cable:     V.35 DCE
Lower DTR:   Disabled
Default Window:     2         SVC idle:  30 seconds
National Personality: GTE Telenet (DTE)
PVC             low: 1    high: 1
Inbound         low: 0    high: 0
Two-Way         low: 2    high: 64
Outbound        low: 0    high: 0
Throughput Class in bps Inbound:  2400
Throughput Class in bps Outbound: 2400

X.25 National Personality Configuration

Request Reverse Charges:  on  Accept Reverse Charges:     on
Frame Extended seq mode:  off Packet Extended seq mode:   off
```

```
Incoming Calls Barred:    off Outgoing Calls Barred:      off
Throughput Negotiation:   on  Flow Control Negotiation:   on
Suppress Calling Addresses: off  DDN Address Translation:  off
Truncate Called Addresses:  off
Number of digits to truncate called addresses to:  2
Call Request Timer:     20 decaseconds
Clear Request Timer:    18 decaseconds (1 retries)
Reset Request Timer:    18 decaseconds (1 retries)
Restart Request Timer:  18 decaseconds (1 retries)
Min Recall Timer        10 seconds
Min Connect Timer       90 seconds
Collision Timer          5 seconds
T1 Timer:  4.00 seconds     N2 timeouts: 20
T2 Timer:  2.00 seconds     DP Timer:    500 milliseconds
Standard Version:     1984      Network Type: CCITT
Disconnect Procedure: passive
Window Size     Frame:   7      Packet:  2
Packet Size     Default: 128    Maximum: 256

X.25 protocol configuration

No protocols defined

X.25 PVC configuration

No PVCs defined

X.25 address translation configuration
No address translations defined
```

detailed

Lists the value of all the default parameters that the **national set** command modifies. Descriptions of the screen display are listed in the **national set** command described later in this chapter.

Example: `list detailed`

```
X.25 National Personality Configuration

Request Reverse Charges: on   Accept Reverse Charges:     on
Frame Extended seq mode: off  Packet Extended seq mode:   off
Incoming Calls Barred:   off  Outgoing Calls Barred:      off
Throughput Negotiation:  on   Flow Control Negotiation:   on
Suppress Calling Addresses: off  DDN Address Translation: off
Truncate called address: off
Number of digits to truncate address to: 2
Call Request Timer:     20 decaseconds
Clear Request Timer:    18 decaseconds (1 retries)
Reset Request Timer:    18 decaseconds (1 retries)
Restart Request Timer:  18 decaseconds (1 retries)
Min Recall Timer        10 seconds
Min Connect Timer       90 seconds
Collision Timer         10 seconds
T1 Timer: 4.00 seconds     N2 timeouts: 20
T2 Timer: 2.00 seconds     DP Timer:    500 milliseconds
Standard Version:     1984      Network Type: CCITT
Disconnect Procedure: passive
Window Size     Frame:   7      Packet:  2
Packet Size     Default: 128    Maximum: 256
```

protocols

Lists all the defined protocol configurations. See “Add” on page 29-16 for a description of the parameters.

Example: `list protocols`

```
X.25 protocol configuration

Protocol       Window         Packet-Size        Idle    Max
 Number         Size          Default Maximum    Time    VCs

 0(IP)           2              128   256         30      4

QLLC Protocols

Protocol     Packet        Idle    Response    Reverse_Charges   Max       Station
 Number    Window MaxSize  Time   Timer Count  Accept Request   Message      Type

26(DLSW)     7      256     30     15    255      N      N        2048       PEER
          CUD : [C3 01 00 00 52 54 50 ]
```

pvc

Lists all the defined PVCs.

Example: `list pvc`

```
X.25 PVC configuration

Prtcl    X.25 Address    Active Enc    Window    Pkt_len   Pkt_chan
0          8383838383        CC           4      1024       3
```

summary

Lists all the values established by the **set** and **enable** commands. These values modify the X.25 configuration.

Example: `list summary`

```
X.25 Configuration Summary

Node Address:       313131
Max Calls Out:           4
Inter-Frame Delay:       0    Encoding:  NRZ
Speed:           64000        Clocking:  Internal
MTU:              2048        Cable:     V.35 DCE
Lower DTR:   Disabled
Default Window:     2         SVC idle:  30 seconds
National Personality: GTE Telenet (DTE)
PVC             low: 1    high: 1
Inbound         low: 0    high: 0
Two-Way         low: 2    high: 64
Outbound        low: 0    high: 0
Throughput Class in bps Inbound:  2400
Throughput Class in bps Outbound: 2400
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

Chapter 30. Monitoring the X.25 Network Interface

This chapter describes the X.25 console commands and includes the following sections:

Accessing the Interface Console Process

To monitor information related to the X.25 network interface, access the interface console process as follows:

1. At the OPCON prompt, enter **talk 5**. For example:

   ```
   * talk 5
   +
   ```

   The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter GWCON, press **Return** again.

2. At the GWCON prompt, enter the **configuration** command to see the protocols and networks for which the router is configured. For example:

   ```
   + configuration
   ```

   See page "Configuration" on page 6-6 for sample output of the **configuration** command.

3. Enter the **network** command and the number of the X.25 interface.

   ```
   + network 2
   X.25>
   ```

   The X.25 console prompt is displayed on the console. You can then view information about the X.25 interface by entering the X.25 console commands.

X.25 Console Commands

This section summarizes and explains all the X.25 console commands. The X.25 console commands allow you to view the parameters and statistics of the interfaces and networks that transmit X.25 packets. Console commands display configuration values for the physical, frame, and packet levels. You also have the option of viewing the values for all three protocol levels at once.

Enter the X.25 console commands at the `X.25>` prompt. Table 30-1 on page 30-2 shows the commands.

Table 30-1. X.25 Console Command Summary

Console Command	Function
? (Help)	Lists all the X.25 console commands or lists the options associated with specific commands.
List	Lists individual PVC or SVC statistics and general information.
Parameters	Displays the current parameters for any level of the X.25 configuration.
Statistics	Displays the current statistics for any level of the X.25 configuration.
Exit	Exits the X.25 console process and returns to the GWCON process.

? (Help)

Use the *? (help)* command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
List
Parameters
Statistics
Exit
```

List

Use the **list** command to display the current active PVCs and SVCs.

Syntax: list pvcs
svcs

pvc

Displays the configured permanent virtual circuits.

Example: **list pvcs**

svc

Displays the active switched virtual circuits.

Example: **list svcs**

```
LCN/  Destination  Originate Transmits  Protocol    Totals
State      Address     Call   Queued  Encapsulated Xmts Rcvs Resets

13 D   898280077113   YES     0          IP      8943  261    1
20 D   898280077114    NO     0          IP       943   43    0
42 P   898280077116   YES     6          IP         0    0    0
23 C   898280077117   YES     0          IP      3054  110    0


D  - Data Transfer      P  - Call Progressing
C  - Call Clearing
```

Parameters

Use the **parameters** command to display the current parameters for any level of the X.25 configuration.

Syntax: parameters
- all
- frame
- packet
- physical

all
Displays the parameters for the packet, frame, and physical levels.

Example: **parameters all**

frame
Displays the parameters for the frame level.

Example: **parameters frame**

```
Frame Layer Parameters:
Maximum Frame Size  = 262 Maximum Window Size  =     7
Protocol Enabled    = YES Equipment Type       =   DTE
T1 Retransmit Timer =   4 T2 Acknowledge Timer =     2
N2 Retry Counter    =  20 Disconnect Procedure =   PASSIVE
Disconnect Timer    = 500 Network Type         =   GTE
Protocol Options: Inhibit Idle RRs No   MOD 128 NO   Enable SARM  NO
```

packet
Displays the parameters for the packet level.

Example: **parameters packet**

```
Packet Layer Parameters:
Default Packet Size = 128  Maximum Packet Size =   256
Log 2 Packet size   =   2  Acknowledge Delay  =     0
Layer Enabled       = YES Default Window Size =     2
Lowest SVC          =   1  Highest SVC        =    64
Lowest PVC          =   0  Highest PVC        =     0
T20 (Restart)       = 18  R20 (Retry)         =     1
T21 (Call)          =    20
T22 (Reset)         = 18  R22 (Retry)         =     1
T23 (Clear)         = 18  R23 (Retry)         =     1
Network Type        = GTE  Equipment Type     =   DTE
Recall Timer        = 10 seconds
Min Connect         = 90 seconds
Collision           = 5 seconds
```

physical
Displays the parameters for the physical level.

Example: **parameters physical**

```
Physical Layer Parameters:
Interface Type        =    V.35

Maximum Frame Size    =    264   InterFrame Delay    =         2
Configured Speed      =      0   Clocking           =  External
Encoding              =    NRZ
Protocol Enabled      =    Yes
```

Statistics

Use the **statistics** command to display the current statistics of any level of the X.25 configuration.

Syntax: statistics
- all
- frame
- packet

physical

all

Displays the statistics for the packet, frame, and physical levels.

Example: **`statistics all`**

frame

Displays the statistics for the frame level.

Example: **`statistics frame`**

```
Frame Layer Counters:      Received            Transmitted
Information Frames              0                      0
RR Command                      0                      0
RR Response                     0                      0
RNR Command                     0                      0
RNR Response                    0                      0
REJ Command                     0                      0
REJ Response                    0                      0
SABM                            0                     71
SABME                           0                      0
UA                              0                      0
DISC                            0                      0
DM                              0                      0
FRMR                            0                      0
Total Bytes                     0                      0
T1 Timeouts   0   T2 Timeouts         0   N2 Timeouts   1
Bad Address   0   Unsolicited F-Bit   0   Invalid Ctl   0Frame Layer Miscellaneous:
Queued Output Frames = 0 Protocol Layer State = Link Setup
Send Sequence N(S)   = 0 Receive Sequence N(R)= 0
```

packet

Displays the statistics for the packet level.

Example: **`statistics packet`**

```
Packet Counters:           Received            Transmitted
Call Request                    0                      0
Call Accepted                   0                      0
Clear Request                   0                      0
Clear Confirm                   0                      0
Interrupt Request               0                      0
Interrupt Confirm               0                      0
RR Packet                       0                      0
RNR Packet                      0                      0
REJ Packet                      0                      0
Reset Request                   0                      0
Reset Confirm                   0                      0
Restart Request                 0                      0
Restart Confirm                 0                      0
Diagnostic                      0                      0
Data Packet                     0                      0
Data Bytes                      0                      0
Buffers Queued                  0                      0
Invalid Packets Received =      0
Switched Circuits Opened =      0
```

physical

Displays the statistics for the physical level.

Example: **`statistics physical`**

```
X.25 Physical Layer Counters:
 Rx Bytes                        0   Tx Bytes                    0

 Adapter cable:          V.35 DTE

 V.24 circuit: 105 106 107 108 109 125 141
  Nicknames:    RTS CTS DSR DTR DCD RI  LL
  PUB 41450:    CA  CB  CC  CD  CF
  State:        ON  ON  ON  ON  ON  OFF OFF
 Line speed:              unknown
 Last port reset:         12 minutes, 21 seconds ago

Input frame errors:
 CRC error                    0   alignment (byte length)         0
 missed frame                 0   too long (>     0 bytes)        0
 aborted frame                0   DMA/FIFO overrun                0
 L & F bits not set           0Output frame counters:
 DMA/FIFO underrun errors      0   Output aborts sent          0
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

X.25 Network Interfaces and the GWCON Interface Command

While X.25 interfaces have their own console processes for monitoring purposes, the router also displays complete statistics for installed network interfaces when you use the **interface** command from the GWCON environment. (For more information on the **interface** command, refer to Chapter 6, The GWCON (Monitoring) Process and Commands).

Statistics Displayed for X.25 Interfaces

The following statistics display when you run the **interface** command from the GWCON environment for X.25 interfaces:

```
+interface 2
 Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
 2  2   X25/0        81640   5C          0          0            0

   X.25 MAC/data-link on SCC Serial Line interface
 Interface State:  DCD  CTS   Packet Layer   Frame Layer
                   OFF  OFF       DOWN          DOWN

   Packet Counters:           Received             Transmitted
    Data Packet                     0                      0
    Data Bytes                      0                      0
    Buffers Queued                0                        0
    Invalid Packets Received =      0
    Switched Circuits Opened =      0

    Frame Layer Counters:      Received             Transmitted
    Information Frames              0                      0

   X.25 Physical Layer Counters:
     Rx Bytes                       0   Tx Bytes              0

     Adapter cable:         Generic DTE  RISC Microcode Revision:       2

      V.24 circuit: 105 106 107 108 109 125 141
      Nicknames:    RTS CTS DSR DTR DCD RI  LL
      PUB 41450:    CA  CB  CC  CD  CF
      State:        ON  OFF OFF ON  OFF OFF OFF

     Line speed:              unknown
     Last port reset:         23 minutes, 48 seconds ago

     Input frame errors:
    CRC error                    0   alignment (byte length)         0
    missed frame                 0   too long (>     0 bytes)        0
    aborted frame                0   DMA/FIFO overrun                0
    L & F bits not set           0
   Output frame counters:    DMA/FIFO underrun errors       0   Output aborts sent          0

   Interface buffer pool: Total = 30, Free = 30
```

The following table describes the interface statistics:

Nt	Global interface number
Nt '	Reserved for future dial circuit use
Interface	Interface name and number (within interfaces of the same type)
CSR	COMM and Status Registers address
Vec	Interrupt vector
Self-Test Passed	Number of times self-test succeeded
Self-Test Failed	Number of times self-test failed
Maintenance Failed	Number of maintenance failures
Interface state	Display the current state of the input modem control signals, the packet layer (X.25 layer 3), and the frame layer (X.25 layer 2).
Packet Counters	Provides statistics on packets received and transmitted.
Data Packets	Displays the number of data packets the interface transmits receives on the network
Data Bytes	Displays the number of data bytes the interface transmits receives on the network.

Buffers Queued	Displays the number of buffers currently queued for transmission over the network. These may be frame or packet layer supervisory messages as well as forwarder packets.
Invalid Packets Received	Displays the number of invalid X.25 packets received from the network.
Switched Circuits Open	Displays the number of switched circuits currently open.
Frame Layer Counters	Provides statistics generated from Frame Layer counters.
Information Frames	Displays the number of X.25 Information frames the interface has transmitted and received.
X.25 Physical Layer Counters	Provides statistics generated from Physical Layer counters.
RX Bytes	Display the number of bytes received by the Physical layer.
TX Bytes	Displays the number of bytes transmitted by the Physical layer.
V.24 circuit Nicknames State	The circuits, control signals, pin assignments and their state (ON or OFF). Note: The symbol `- - -` in console output indicates that the value or state is unknown.
Line speed	The transmit clock rate.
Last port reset	The length of time since the last port reset.
Input frame errors:	
CRC error	The number of packets received that contained checksum errors and as a result were discarded.
Alignment	The number of packets received that were not an even multiple of 8 bits in length and as a result were discarded.
Too short	The number of packets that were less than 2 bytes in length and as a result were discarded.
Too long	The number of packets that were greater than the configured size, and as a result were discarded.
Aborted frame	The number of packets received that were aborted by the sender or a line error.
DMA/FIFO overrun	The number of times the serial interface card could not send data fast enough to the system packet buffer memory to receive them from the network.
Missed frame	When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.
L & F bits not set	On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the `L & F bits not set` counter is incremented, and the buffer is cleared for reuse. **Note:** It is unlikely that the `L & F bits not set` counter will be affected by traffic.
Output frame counters:	
DMA/FIFO underrun errors	The number of times the serial interface card could not retrieve data fast enough from the system packet buffer memory to transmit them onto the network.

Output aborts sent	The number of transmissions that were aborted as requested by upper-level software.

Chapter 31. Using and Configuring Frame Relay Interfaces

This chapter describes the Frame Relay configuration commands and includes the following sections:

Frame Relay Overview

The Frame Relay (FR) protocol is a method of transmitting internetworking packets by combining the packet switching and port sharing of X.25 with the high speed and low delay of time division multiplexing (TDM) circuit switching. FR allows you to connect multiple LANs to a single high-speed (1.54 Mbps) WAN link with multiple point-to-point permanent virtual circuits (PVCs). FR offers the following features:

- *High throughput and low delay.* Utilizing the *core aspects* (error detection, addressing, and synchronization) of the Link Access Protocol, D-Channel (LAPD) datalink protocol, FR eliminates all network layer (Layer 3) processing. By using only the core aspects, FR reduces the delay of processing each frame.
- *Congestion detection.* Upon receiving Backward Explicit Congestion Notification (BECN) or a Forward Explicit Congestion Notification (FECN) , the router initiates a controlled slowdown of traffic, thereby avoiding a complete FR network shutdown.

  The router can also initiate a slowdown of traffic when it receives a Consolidated Link Layer Management (CLLM) congestion message. CLLM is an optional part of the Frame Relay standards that provides additional management information about the operation of the frame relay network to attaching DTEs.
- *Circuit access and control.* As the router dynamically learns about the availability of non-configured circuits (orphan circuits), you can control access to those new circuits.
- *Network management option.* As your network requires, the FR protocol can operate with or without a local network management interface.
- *Multiplexing protocols.* Using one PVC to pass multiple protocols.
- *Data compression.* that supports the FRF.9 standard. See Chapter 19, "The Data Compression Subsystem" on page 19-1 for details.

FR provides no error correction or retransmission function. To provide error–free end-to-end transmission of data, FR relies on the intelligence of the host devices.

Frame Relay Network

The FR network consists of the FR backbone (consisting of FR switches provided by the FR carrier) providing the FR service. The router functions as the FR connection device. The router encapsulates FR frames and routes them through the network based on a Data Link Connection Identifier (DLCI). The DLCI is the medium access control (MAC) address that identifies the PVC between the router and the FR destination device. For example, in Figure 31-1, a packet destined to go from router B to router D would have a DLCI of 19 to reach router D; however, a packet destined to go from router D to router B would have a DLCI of 16.

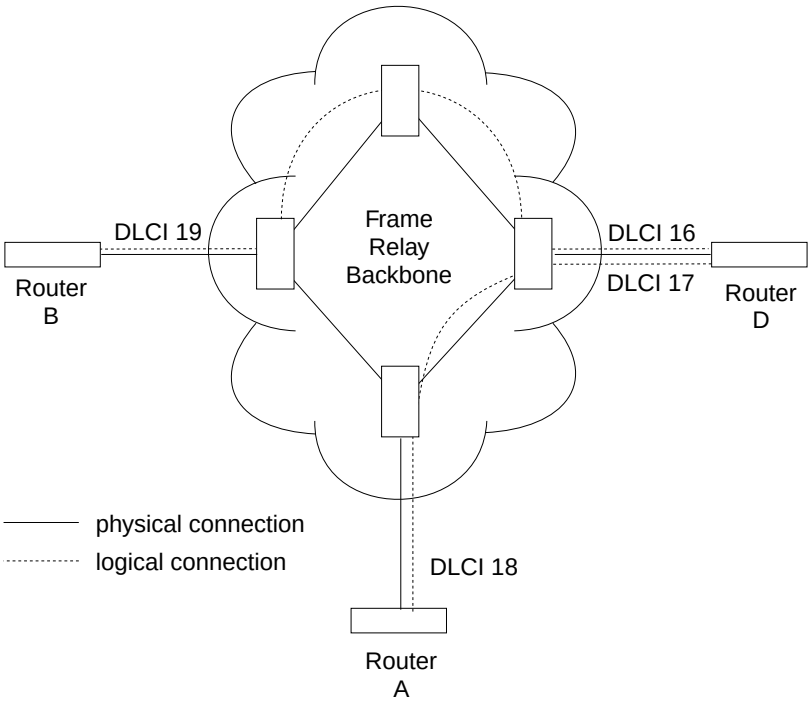


Figure 31-1. DLCIs in Frame Relay Network

A DLCI can have either local or global significance. Local DLCIs are significant at the point of entry to the network, but global DLCIs are significant throughout the network. To the user, however, the DLCI that the router uses to route a packet is the DLCI that the user associates with the frame's global or local destination. DLCIs are configured through the FR configuration process or learned through FR management.

A Frame Relay network has the following characteristics:

- Transports frames transparently The network can modify only the DLCI, congestion bits, and frame check sequence. High-Level Data Link Control (HDLC) flags and zero bit insertion provide frame delimiting, alignment, and transparency.
- Detects transmission, format, and operational errors (frames with an unknown DLCI)
- Preserves the ordering of frame transfer on individual PVCs
- Does not acknowledge or retransmit frames

Frame Relay Interface Initialization

If a Local Management Interface (LMI) is enabled, the FR interface is active when a successful exchange of LMI frames occurs between the router and the FR switch; however, no data can be received from or transmitted to another router until an LMI status message indicates that the PVC status for the DLCI to the other router is active. Also, there are instances where the FR interface state is tied to PVC states and the interface does not come up even if LMI exchanges are successfully occurring (for additional information, see “Configuring PVC States to Affect the Frame Relay Interface State” on page 31-4).

PVC status appears for all PVCs as either active or inactive. An active PVC has a completed connection to an end system. An inactive PVC does not have a completed connection to an end system because either an end system or an FR switch is off-line.

For example, in Figure 31-2 router B has a configured PVC to router D. Router B is successfully interacting with FR management through FR switch B. Because either another FR switch is down or the end system is down, the end-to-end PVC connection is not established. Router B receives an inactive status for that PVC.

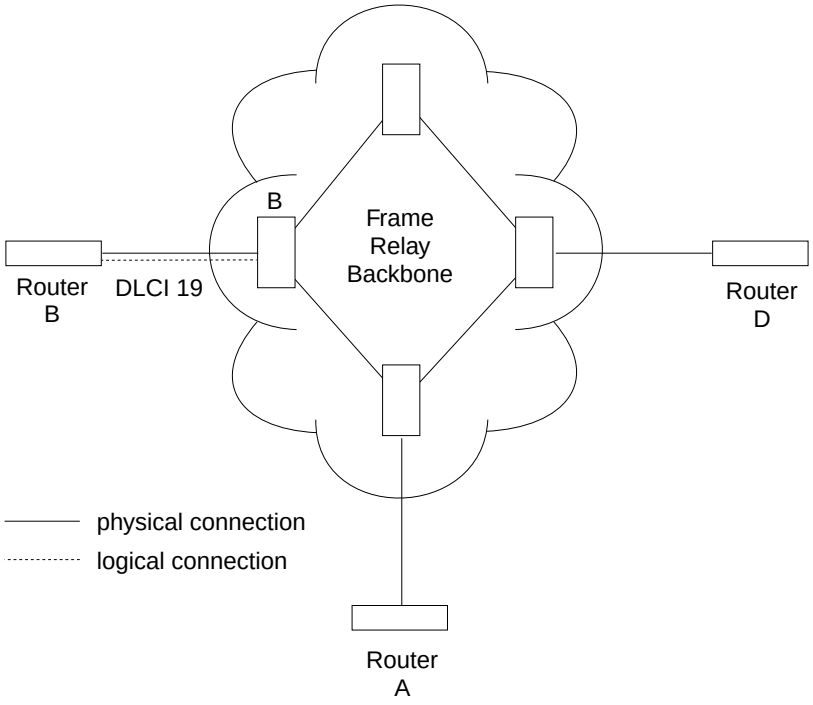


Figure 31-2. DLCIs in Frame Relay Network

When the Local Management Interface (LMI) is disabled, the FR interface is running on a serial line and a DTE cable is being used, the FR protocol asserts the DTR and RTS modem control signals. (The Control signal is asserted for X.21). The FR interface goes up once the DSR, CTS, and DCD modem control signals are on. (When X.21 is used, the FR interface goes up once the Indication modem control signal is on.) The FR interface is down or in the testing state if either DSR, CTS, or DCD are off or, when X.21 is used, the Indication signal is off. Therefore, you need to ensure that the modem, modem eliminator, or DSU that is used drops one or more of these signals when the physical connection to the FR switch or the other FR DTE (if configured for FR DTE to DTE connectivity) is lost.

Orphan Circuits

An *orphan circuit* is any PVC that is not configured for your router but is learned indirectly through the actions of the network management entity. For example, Figure 31-3 assumes that router B has a configured PVC to router D, but none to router A. Router A configures a PVC to router B. Router B would then learn about the PVC to router A from LMI messages and classify it as an orphan.

Orphan circuits are treated the same as configured circuits except that you may enable or disable their use with the **enable orphan-circuit** and **disable orphan-circuit** commands.

By disabling orphan circuits, you add a measure of security to your network by preventing any unauthorized entry into your network from a non-configured circuit. By enabling orphan circuits, you allow the router to forward packets over circuits you did not configure. Packets that would normally be dropped are now forwarded.

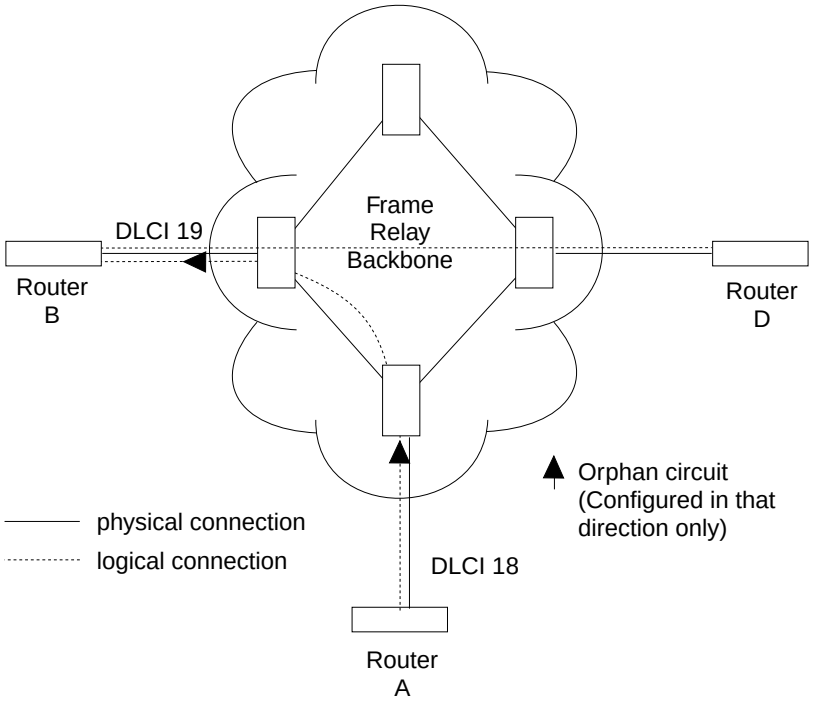


Figure 31-3. Orphan Circuit

Configuring PVC States to Affect the Frame Relay Interface State

You can control the operation of your Frame Relay interface by

1. Enabling the “No-PVC” feature or
2. Configuring “required PVCs” or
3. Configuring “required PVC groups.”

By enabling the Frame Relay “No-PVC” feature, the Frame Relay interface becomes inactive when there are no active PVCs on the interface. If at least one PVC is active, the Frame Relay interface becomes active when a successful LMI exchange occurs between the router and the FR switch.

You can configure a PVC as a “required PVC.” If a PVC is required but not in a group, the Frame Relay interface becomes inactive when the PVC becomes inactive. When the PVC becomes active, the interface is activated following a successful exchange of LMI frames between the router and the Frame Relay switch.

If multiple PVCs are required and are not in a PVC group, the interface is not activated until all required PVCs are active.

If a required PVC belongs to a PVC group, the Frame Relay interface becomes inactive when all PVCs in the PVC group are inactive. If at least one PVC in the group is active, the interface becomes active following a successful exchange of LMI frames between the router and the FR switch. If there are multiple PVC groups, the interface does not become active until at least one PVC *in each group* is active.

A “required PVC group” is a group of circuits associated by name, where “name” is the name of the required PVC group.

These features can be used with WAN Reroute so that an alternate link can be brought up if all PVCs, required PVCs, or a group of PVCs become inactive on the primary FR link.

Frame Relay Frame

An FR frame consists of a fixed size address field with variable sized encapsulated user data. Figure 31-4 illustrates a Frame-Relay frame format.

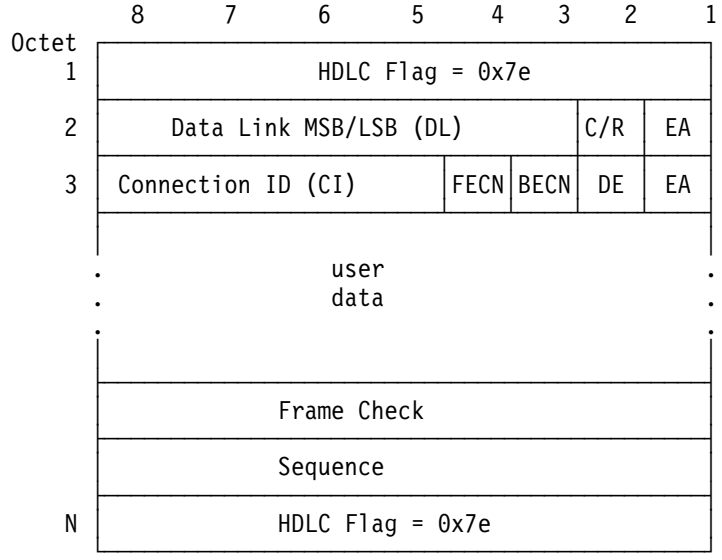


Figure 31-4. Frame-Relay Frame Format

HDLC Flags

Located in the first and last octet, these flags indicate the beginning and end of the frame.

Data Link Connection Identifier (DLCI)

This 10-bit routing ID resides in bits 3 to 8 of octet 2 and bits 5 to 8 of octet three. The DLCI is the MAC address of the circuit. The DLCI allows the user and network management to identify the frame as being from a particular PVC. The DLCI enables multiplexing of several PVCs over one physical link.

Command/Response (C/R)

This field's use is not defined within the Frame-Relay standards and the field is passed transparently across the network.

Extended Address

This version of FR does not support extended addressing.

Forward Explicit Congestion Notification (FECN)

The FR backbone network sets this bit to 1 to notify the user receiving the frame that congestion is occurring for the PVC in the direction the frame is being sent. You can configure the device to slow down data transmission in the direction from which it receives a FECN using the **enable throttle-transmit-on-fecn** command. You can also set the BECN bit in data frames sent to the originator of the FECN using the **enable notify-fecn-source** command.

APPN High Performance Routing (HPR) uses detection of this bit set to allow Rapid Transport Protocol's adaptive rate-based flow and congestion control algorithm to adjust the data send rate. This algorithm prevents traffic bursts and congestion, maintaining a high level of throughput.

Backward Explicit Congestion Notification (BECN)

The FR backbone network sets this bit to 1 to notify the user that the frames sent by this router for this PVC have encountered congestion. The router then initiates a *throttle down* to a rate equal to or less than the user-defined CIR when CIR or congestion monitoring are enabled. The CIR for a PVC is supplied by the FR service provider and is configured using the **add permanent-virtual-circuit** command.

Discard Eligibility (DE)

The Frame Relay network may discard transmitted data exceeding CIR on a PVC. The DE bit can be set by the router to indicate that some traffic should be considered discard eligible. If appropriate, the Frame Relay network will discard frames marked as discard eligible which may allow frames that are not marked discard eligible to make it through the network. To identify traffic that is discard eligible:

1. Configure BRS on the Frame Relay interface and any FR circuits that has traffic that you are making discard eligible.
2. Assign a protocol or filter to a BRS traffic class using the **assign** command. You specify whether the DE bit should be set on for this protocol or filter traffic.

User Data

This field contains the protocol packet being transmitted. This field can contain a maximum of 8188 octets; however, the frame check sequence (FCS) can effectively detect errors only on a maximum of 4096 octets of data. The protocol data is preceded by a Frame Relay encapsulation header as defined in RFC 1490.

Frame Check Sequence

This field is the standard 16-bit cyclic redundancy check (CRC) that HDLC and LAPD frames use. This field detects bit errors occurring in the bits of the frame between the opening flag and FCS.

Frame Forwarding over the Frame Relay Network

When the FR protocol receives a packet for encapsulation, it compares the packet's network address to the entries in the Address resolution Protocol (ARP) cache. If the ARP cache contains the DLCI number that matches the network address, the FR protocol encapsulates that packet into a frame and transmits the frame over its specified local DLCI. If the ARP cache does not contain a match, the FR protocol sends out an ARP request over all configured PVCs on the interface. When the appropriate end-point responds with an ARP response, the FR protocol adds its local DLCI that received the ARP response to the ARP cache. Subsequent data packets directed to the same network address are then encapsulated into a frame and sent out over its local DLCI.

Protocol Addresses

Protocol addresses can be either mapped statically to FR network PVC addresses or discovered dynamically through Inverse ARP or ARP. (For more information on ARP and Inverse ARP, see the *Protocol Configuration and Monitoring Reference*.) Either method is protocol-dependent as illustrated in Table 31-1.

Note: Static protocol addresses are also referred to as static ARP entries. A static ARP entry is added to the configuration with the **add protocol-address** command.

Table 31-1. Protocol Address Mapping

Protocol Type	ARP and Inverse ARP Usage	Static Mapping	PVC Configured at Protocol Configuration
AP2	Yes	Yes	No
IP	Yes	Yes	No
IPX	Yes	Yes	No
Banyan VINES	No	No	No
DNA IV	Yes	Yes	No
OSI*	No	No	Yes

* You must configure OSI at the protocol level to map the protocol address to the FR PVC.

Multicast Emulation and Protocol Broadcast

Multicast emulation is an optional feature that allows protocols requiring multicast such as ARP to function properly over the FR interface. With multicast emulation, a multicast frame is transmitted on each active PVC. By using the **enable** and **disable multicast** commands, you can turn this feature on or off. Protocols that utilize multicast are AP2, ARP, Banyan VINES, DNA4, IP, and IPX.

Protocol broadcast is another optional feature that allows the IP RIP protocol to function properly over the FR interface. By using the **enable protocol-broadcast** and **disable protocol-broadcast** commands, you can turn this feature on or off.

Frame Relay Network Management

The supplier of the FR network backbone provides FR network management. It is management's responsibility to provide FR end-stations (routers) with status and configuration information concerning PVCs available at the interface.

The FR protocol supports the ANSI T1.617 Annex D, ITU-T Q.933 Annex A (also referred to as CCITT Q.933 Annex A), and the Interim Local Management Interface (LMI) management entities. You can turn these entities on or off using the **enable** and **disable** LMI configuration commands. Specifically, FR network management provides the following information:

- Notification of additional PVCs (orphans) and whether they are active or inactive, or notification of any PVC deletions.
- Notification of the availability of a configured PVC. The availability of a PVC is indirectly related to the successful participation of the PVC end-point in the *heartbeat polling* process, which is detailed in "Link Integrity Verification Report" on page 31-9.
- Verification of the integrity of the physical link between the end-station and network by using a *keep alive* sequence number interchange.

Although the FR interface supports network management, it is not necessary for management to run on the FR backbone for the interface to operate over the FR backbone. For example, you may want to disable management for back-to-back testing.

Management Status Reporting

Upon request, FR management provides two types of status reports, a full status report and a link integrity verification report. A full status report provides information about all PVCs the interface knows about. A link integrity verification report verifies the connection between a specific end station and a network switch. All status inquiries and responses are sent over DLCI 0 for ANSI T1.617 Annex D and ITU-T Q.933 Annex A, or DLCI 1023 for interim LMI management.

Full Status Report

When the FR interface requires a full status report, the router's FR protocol sends a status enquiry message to the FR network backbone requesting a full status report. A status enquiry message is a request for the status of all PVCs on the interface. Upon receiving this request, FR management must respond with a full status report consisting of the link integrity verification element and a PVC status information element for each PVC. (See "Link Integrity Verification Report" on page 31-9.)

The PVC status information element contains the following information: the local DLCI number for the particular PVC; the state of the PVC (active or inactive); and whether the PVC is new or an existing PVC that management already knows about.

Note: The number of PVCs supplied at the FR interface is restricted by the network frame size and the amount of individual PVC information elements that can fit into a full status report. For example, 202 is the maximum number of PVCs for a network with a 1K frame size.

Link Integrity Verification Report

The link integrity verification report, sometimes referred to as *heartbeat polling*, contains the link integrity verification element. This element is where the exchange of the send and receive sequence numbers takes place. By exchanging sequence numbers, management and the end station can evaluate the integrity of the synchronous link. The send sequence number is the current send sequence number of the message originator. The receiver looks at this number and compares it to the last send sequence number to verify that this number is incrementally correct. The receive sequence number is the last send sequence number that the originator sent out over the interface. It is the receiver's responsibility to place a copy of the send sequence number into the receive sequence number field. This way the originator can ensure that the receiver receives and interprets the frames correctly.

When an end-station fails to participate in this polling process, all remote end-stations with logically attached PVCs are notified through management's full status report mechanism that the PVC is inactive.

Consolidated Link Layer Management (CLLM)

CLLM is an optional FR management function that is not widely supported by the industry but it has been adopted by some Frame Relay switch manufacturers. CLLM provides some of the same management information provided by LMI, in particular, outage notification. CLLM's main use is to provide asynchronous congestion notification to attaching devices. A single CLLM message may indicate outage or congestion for multiple PVCs. The Frame Relay protocol supports the following standards for CLLM: ANSI T1.618, ITU-T (CCITT) Q.922 Annex A, and ITU-T (CCITT) X.36 Annex C.

Frame Relay Data Rates

This section introduces data rates for Frame Relay permanent virtual circuits (PVCs).

Committed Information Rate (CIR)

The CIR is the data rate that the network commits to support for the PVC under normal, uncongested conditions. Any PVC that is configured or is learned is provided a CIR (by the FR service provider). The CIR is a portion of the total bandwidth of the physical link of either 0 or between 300 bps and 2M reserved for the PVC. 64 Kbps or a single DS0 channel is most common. You define the CIR with the **add permanent-virtual-circuit** or the **change permanent-virtual-circuit** configuration command. You can also dynamically change the CIR with the **set**

circuit console command. You can also set the default CIR for all Frame Relay circuits on this interface using the **set CIR-defaults** command.

Some Frame Relay switches allow a value of 0 to be configured for CIR. When CIR is equal to 0, little or no bandwidth is reserved in the Frame Relay network backbone for the PVC, and the PVC's traffic uses non-reserved bandwidth.

Orphan Circuit CIR

The router assigns a CIR to orphan circuits based on the CIR defaults configured at the interface level. If you are relying on the orphan circuit to route important data and the CIR, Bc, and Be values from the network provider are different from the values configured at the interface level, it is recommended that you define a PVC instead of an orphan circuit. Doing this, you can assign a CIR that the network commits to support.

Committed Burst (Bc) Size

The *committed burst (Bc) size* is the maximum amount of data (in bits) that the network commits to deliver during a *calculated time (Tc) interval*. The Tc is equal to the Bc divided by the CIR (Tc = Bc / CIR). If you configure 0 for CIR, Tc is equal to Be divided by the line speed (or access rate (AR)) (Tc = Be/AR).

For example, if you set a PVC's CIR to 9600 bps and the committed burst size to 14 400 bits, the time period is 1.5 sec. (14 400 bits / 9600 bps = 1.5 sec). This means that the PVC is allowed to transmit a maximum of 14 400 bits in 1.5 seconds.

This parameter is important because of the relationship between the committed burst size and the maximum frame size. If the maximum frame size in bits is greater than the committed burst size, the network may discard frames whose size exceeds the committed burst size. Therefore, the committed burst size should be greater than or equal to the maximum frame size. It should also equal the burst size set up with the network provider.

Use the **add permanent-virtual-circuit** and **change permanent-virtual-circuit** configuration commands to set the committed burst size. The **set circuit** console command can be used to dynamically change the committed burst size. You can also set the default committed burst size for all Frame Relay circuits on this interface using the **set CIR-defaults** command.

The device assigns orphan circuits a committed burst size based on the default you set with the set CIR-defaults command. If you configure 0 for CIR, then the committed burst (Bc) size also equals 0.

Excess Burst (Be) Size

The *excess burst (Be) size* is the maximum amount of uncommitted data the router can transmit on a PVC in excess of the Bc during the Tc (Tc = Bc / CIR) when CIR and Bc are nonzero. When CIR = 0, Tc = Be / AR where the access rate (AR) is the line speed.

The network delivers this excess data with a lower probability of success than committed burst size data. Set the Be to a value greater than zero only if you are willing to accept the risk of discarded data and its effect on higher-layer protocol performance. The Be should equal the value set up with the network provider.

Use the **add permanent-virtual-circuit** command or the **change permanent-virtual-circuit** command during frame-relay configuration to set the excess burst size. You can also use the **set circuit** console command to dynamically change the excess burst size. Orphan circuits will receive a default excess burst size equal to the value set in the **set CIR-defaults** command. If you configure 0 for CIR, then you must configure a nonzero value for the excess burst (Be) size. You can also set the default excess burst size for all Frame Relay circuits on this interface using the **set CIR-defaults** command.

Line Speed

The *line speed* is the interface's line speed.

The FR interface's line speed is configured using the **set line-speed** configuration command. The line speed must be configured when internal clocking is used. However, it is recommended that you configure a line speed for external clocking since the router uses the line speed as the maximum information rate when congestion monitoring is enabled. Also some of the protocols use an interface's configured line speed when calculating a route's cost.

The line speed is not configurable on a Frame Relay dial circuit interface. If the dial circuit is mapped to an ISDN base interface, 64Kbps is used as the line speed.

If the dial circuit is mapped to a V.25bis base interface, the line speed of the V.25bis interface is used for the FR dial circuit.

Minimum Information Rate

The *minimum information rate (IR)* is the minimum data rate for a PVC that the router throttles down to when it is notified of congestion. You set the minimum IR as a percentage of CIR using the **set ir-adjustment** configuration command. It can be dynamically changed using the **set ir-adjustment** console command. If you configure CIR equal to 0, the minimum IR is 1500 bps.

Maximum Information Rate

The *maximum information rate* is the maximum data rate at which the router transmits for a PVC. If the CIR monitoring feature is enabled and CIR and Bc are nonzero, the maximum information rate is calculated using CIR, Bc, and Be as follows:

```
( Bc + Be ) * (Bc / CIR)
```

If the CIR monitoring feature is enabled and CIR and Bc are configured equal to 0, the maximum information rate is equal to the excess burst size (Be).

If the CIR monitoring feature is not enabled the maximum information rate is equal to the line speed.

Variable Information Rate

The *variable information rate* (VIR) ranges from the configured minimum IR to the calculated maximum IR when the CIR monitoring or congestion monitoring features are enabled. The VIR is gradually decreased down to the minimum information rate when the router is notified of congestion on a circuit and is gradually increased to the maximum information rate when the router stops receiving congestion notifications. Using the **set ir-adjustment** configuration command, you configure

the percentage of the information rate by which the VIR should decrease when the router is notified of congestion. You also use this command to configure the percentage of the information rate by which the VIR should be gradually increased when the congestion ends.

To avoid impulse loading of the network, the router initially sets the VIR to CIR when the PVC becomes active. If you configure 0 for CIR, VIR is initially set to excess burst (Be) times the MIR adjustment percentage. For example, if Be is set to 64 000 and the MIR adjustment percentage is set to 25%, then the initial VIR would be equal to 16 000 bps.

The VIR can actually exceed the maximum value in one case. If the length of a frame in bits is greater than the maximum IR, Frame Relay transmits the frame anyway.

Circuit Congestion

Circuit congestion occurs for one of the following reasons:

- The sender is transmitting faster than the allowable throughput
- The receiver is too slow when processing the frames
- An intermediate backbone link is congested, resulting in the sender transmitting faster than the available throughput allows.

When circuit congestion happens, the network must drop packets and/or shut down.

In response to circuit congestion, the router implements a *throttle down*, which is a step-wise slowing of packet transmission to the configured minimum IR. Throttle down occurs during the following conditions:

- Circuit congestion is occurring.
- The router is the sender of frames.
- CIR monitoring or congestion monitoring is enabled.

This section discusses monitoring of Frame Relay data rates and circuit congestion.

CIR Monitoring

CIR monitoring is an optional Frame Relay feature that you can set for each interface to prevent the router from creating congestion conditions in the FR network. CIR monitoring allows the VIR for a PVC to range between the configured minimum and maximum IR.

CIR monitoring is configured with the **enable cir-monitor** configuration command and is disabled by default. CIR monitoring, when enabled, overrides congestion monitoring. You can also dynamically enable and disable CIR monitoring using the **enable cir-monitor** and **disable cir-monitor** console commands.

Congestion Monitoring

Congestion monitoring is an optional feature, set per interface, that allows the VIR of PVCs to vary in response to network congestion. The VIR assumes values between the minimum IR and a maximum IR of the line speed. Congestion monitoring is enabled by default. It can be disabled with the **disable congestion-monitor** configuration command and re-enabled with the **enable**

congestion-monitor command. You can also dynamically enable and disable congestion monitoring using the **enable congestion-monitor** and **disable congestion-monitor** console commands.

CIR monitoring, if enabled, overrides congestion monitoring. If both CIR monitoring and congestion monitoring are disabled, the VIR for each PVC on the interface is set to the line speed and does not decrease in response to network congestion.

Note: Even with compression enabled, the device uses the uncompressed size of frames to determine if the VIR is being exceeded.

Congestion Notification and Avoidance

When congestion occurs, the FR backbone network is responsible for notifying the sender and receiver by sending out a FECN or a BECN signal. FECN and BECN are bits that are set in a frame to notify the DTEs at each end of a PVC that congestion is occurring. FECN indicates that congestion is occurring in the same direction from which the frame was received; the sender is causing the congestion. BECN indicates that the frames sent by this DTE are causing network congestion.

Optionally, the network can use CLLM messages to convey congestion information. CLLM messages are sent only to the congestion source and should be treated similarly to BECN messages by the DTE.

The example in Figure 31-5 on page 31-14 shows a congestion condition at switch B when frames are sent from router X to router Y. The FR backbone network notifies router X that frames it sends are encountering congestion by setting the BECN bit in frames sent to router X. The FR backbone network also notifies router Y that frames it receives encountered congestion by setting the FECN bit.

When the router receives a frame containing BECN, it is the router's responsibility to throttle down the PVC's VIR (variable information rate) if either CIR monitoring or congestion monitoring is enabled. The router does this gradually as it receives consecutive frames with BECN until either the minimum IR is reached or a frame without BECN arrives. As the router receives consecutive frames without BECN, the VIR gradually rises to the maximum IR.

Depending on the operation of the FR network, it may be necessary for the device to throttle down the PVC's VIR when the device receives a FECN to minimize the overall amount of traffic being offered to the network as quickly as possible. Reducing the overall load on the network reduces the number of packets discarded for all PVCs to relieve congestion. Enabling the ***throttle-transmit-on-fecn*** parameter, along with either the CIR or congestion monitoring options, causes the device to treat a FECN like a BECN thus reducing overall FR network congestion when any congestion notification is received. Use the throttle-transmit-on-fecn parameter only in FR networks whose queuing methods do not provide dedicated buffers for both input and output.

Some FR network switches set FECN to indicate congestion but do not set BECN. To provide congestion notification to the source of the congestion, enable the ***notify-fecn-source*** parameter allowing the device to set BECN in frames that it transmits over a PVC on which it has received a FECN. This action provides a signal to the device that is causing the network congestion to throttle down its PVC's VIR.

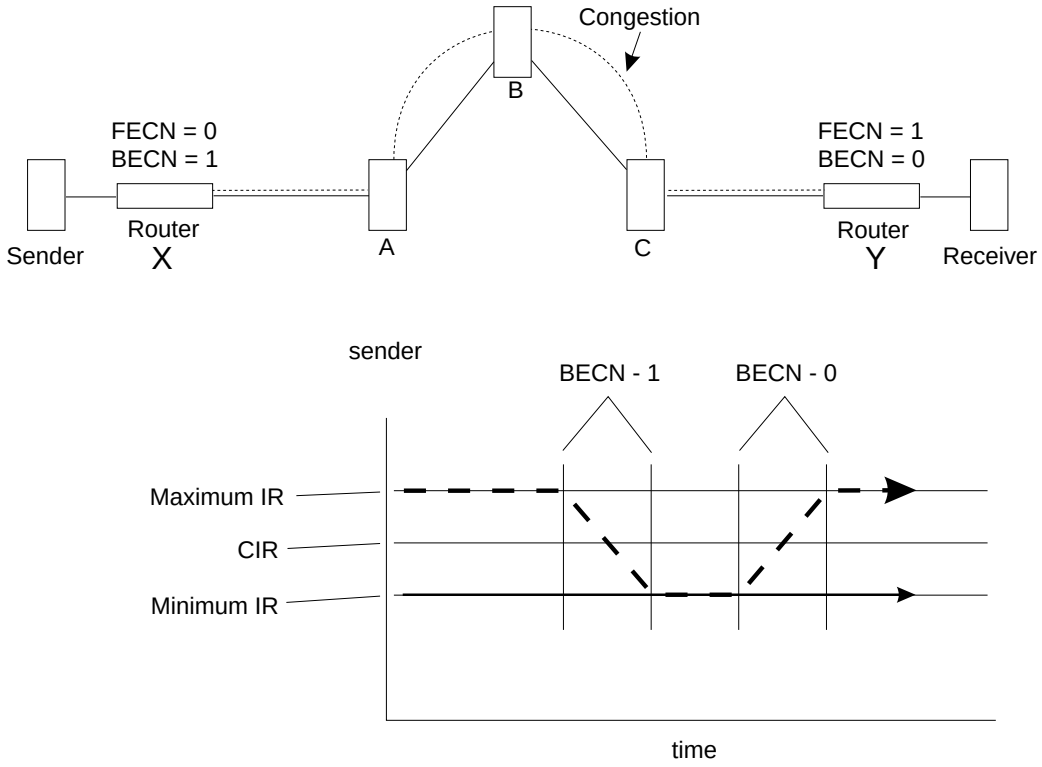


Figure 31-5. Congestion Notification and Throttle Down

Note: If multiple DLCIs are configured between two end-stations when congestion occurs, it is possible that a second DLCI may be used to transmit data at a higher throughput until the congestion condition on the first DLCI is corrected.

Similarly, if the network provider supports CLLM, you can configure Frame Relay to *throttle down* its transmit rate for PVCs contained in a CLLM message. CLLM messages contain a cause code that indicates the type and severity of the problem being reported. The device reacts differently depending on the cause code and the CIR configured for each PVC contained in the CLLM message. When the device receives a CLLM message that indicates:

- A short-term condition, and the configured CIR for the PVC is nonzero, the Frame Relay protocol will throttle the transmit rate for the affected PVCs by the configured IR decrement percentage.
- A long-term condition, the Frame Relay protocol will set the transmit rate for the affected PVCs to the calculated minimum information rate.
- Facility or equipment failure or maintenance action, or if the CIR was configured as zero, the FR protocol will continue to transmit any queued data for the affected PVCs but will not accept any more outgoing packets from the upper layer protocols until the congestion condition is cleared.

Once a CLLM message for a PVC has been received, if the device does not receive any CLLM messages or BECNs within the *Ty* timer period or if a frame without a BECN is received, the device will consider the congestion condition cleared and gradually return the PVC to its configured transmission rates. If you are using CLLM to control congestion, you must not configure DLCI 1007 for any other use.

Bandwidth Reservation over Frame Relay

For information on bandwidth reservation over Frame Relay, refer to Chapter 10, “Using and Configuring Bandwidth Reservation and Priority Queuing” on page 10-1 through Chapter 11, “Monitoring Bandwidth Reservation” on page 11-1.

Displaying the Frame Relay Configuration Prompt

To access the Frame Relay configuration environment:

1. At the OPCON prompt (*), type **talk 6**.
2. At the configuration prompt (Config>), enter the **list devices** command to see a list of interfaces configured on the router.
3. Enter the **network** command to display the Frame Relay configuration prompt. The network number is the number of the Frame Relay interface.

```
Config>network
What is the network number [0] 2
Frame Relay user configuration
FR 2 Config >
```

4. At the Frame Relay interface configuration prompt (FR Config>), use the commands discussed in this chapter to configure Frame Relay parameters.

Frame Relay Basic Configuration Procedure

This section outlines the minimum configuration steps that you are required to perform to get the Frame Relay protocol up and running. If you desire any further configuration information and explanation, refer to the configuration commands described in this chapter.

Note: You must restart the router for new configuration changes to take effect.

- **Select FR management.** The FR Local Management Interface (LMI) protocol defaults to ANSI. You have the option of connecting to a network using the Interim LMI (REV1), ANSI T1.617 Annex D management, or ITU-T/CCITT Q.933 Annex A management. Use the **enable** and **set** commands to enable and set the required management.
- **Add a PVC.** Add any required PVCs that are needed if FR management is disabled or orphan circuits are disabled. If you want to bridge over a FR PVC, or if you want to run APPN over a FR PVC, you also must configure that PVC. Use the **add permanent-virtual-circuit** command.
- **Configure FR destination addresses.** If you are running a protocol such as IP or IPX over the FR interface, and are interconnecting with devices not supporting the Address Resolution Protocol (ARP) or Inverse ARP on FR, use the **add protocol-address** command to add the static protocol and address mapping.
- **Configure Bandwidth Reservation over Frame Relay.** In addition to the basic Frame Relay configuration, which must be done, you can also configure Bandwidth Reservation (an optional feature) over Frame Relay. For information on configuring Bandwidth Reservation, refer to Chapter 10, “Using and Configuring Bandwidth Reservation and Priority Queuing” on page 10-1.
- **Configure Discard Eligibility.** You can configure Discard Eligibility (DE) congestion control using Bandwidth Reservation. For information on configuring

Discard Eligibility, refer to Chapter 10, “Using and Configuring Bandwidth Reservation and Priority Queuing” on page 10-1.

- **Configure Data Compression.** You can configure data compression for Frame Relay. For information on configuring data compression, refer to Chapter 19, “The Data Compression Subsystem” on page 19-1.

Enabling Frame Relay Management

There are three management options under Frame Relay:

- Interim Local Management Interface Revision 1
- ANSI T1.617 Annex D management
- ITU-T/CCITT Q.933 Annex A management.

Frame Relay defaults to ANSI enabled. If you want to change management types, or if you want to re-enable ANSI management, use the following procedure. Enabling management over Frame Relay is a two-step process:

1. Enter the **enable lmi** command at the `FR Config>` prompt to enable management activity.
2. Enter the **set lmi-type** command to select the type of management for the interface.

   See Table 31-2 for details of the management types available using the **set** command.

   An example of how to set these management types is shown after the table. Also, refer to the **enable** and **set** command sections in this chapter for more information.

Table 31-2. Frame Relay Management Options

Command	Options	Description
set	lmi-type rev1	Conforms to LMI Revision 1 (Stratacom s Frame Relay Interface Specification)
set	lmi-type ansi	Conforms to ANSI T1.617 ISDN-DSS1-Signalling Specification for Frame Relay Bearer Service (known as Annex D)
set	lmi-type ccitt	Conforms to Annex A of ITU-T/CCITT Recommendation Q.933 - DSS1 Signalling Specification for Frame Mode Basic Call Control.

Example:

```
enable lmi
set lmi-type ansi
```

Frame Relay Configuration Commands

This section summarizes and then explains the Frame Relay configuration commands. Enter all commands at the `Frame Relay>` prompt.

You must restart the router for new configuration changes to take effect.

Table 31-3. Frame Relay Configuration Commands Summary

Command	Function
? (Help)	Lists the configuration commands or lists any parameters associated with the commands.
Add	Adds PVCs, Required PVC groups, and destination protocol addresses to the Frame Relay interface.
Change	Modifies a PVC or Required PVC group previously defined by the **add** command.
Disable	Disables any enabled Frame Relay features.
Enable	Enables Frame Relay features such as circuit monitoring, management options, multicast, protocol-broadcast, and orphans.
List	Displays the current configuration of the LMI, PVCs, Required PVC groups, HDLC information, and protocol addresses.
LLC	Configures LLC parameters on the Frame Relay interface. These LLC parameters are required when running APPN over the Frame Relay interface.
Remove	Deletes any previously added PVCs, Required PVC groups (if empty), or protocol addresses.
Set	Configures the Frame Relay management options and parameters (N1-parameter, N2-parameter, N3-parameter, P1 parameter, and T1-parameter). Configures the physical-layer parameters for FR serial interfaces. Sets the maximum frame size.
Exit	Exits the Frame Relay configuration and returns to the `Config>` prompt.

Note: In this section, the terms *circuit number* and *PVC* are synonymous with the term DLCI (Data Link Circuit Identifier).

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
Add
Change
Disable
Enable
List
Remove
Set
Exit
```

Example: **`enable ?`**

```
cir-monitor
cllm
compression
congestion-monitor
DN-length-field
lmi
lower-dtr
multicast-emulation
no-pvc
notify-fecn-source
orphan-circuits
protocol-broadcast
throttle-transmit-on-fecn
```

Add

Use the **add** command to add a PVC, Required PVC group, or destination protocol address supported by the Frame Relay interface.

Syntax: add permanent-virtual-circuit . . .
protocol-address . . .
pvc-group . . .

permanent-virtual-circuit

Adds a PVC to the Frame Relay interface beyond the reserved range 0 through 15. The maximum number of PVCs that can be added is approximately 992, but the actual number of PVCs that the interface can support depends on the throughput required for each PVC, the line speed, the type of protocols running on the interface, and the number of local management interface PVC information elements that can fit in the maximum frame size.

Example: **add permanent-virtual-circuit**

```
Circuit Number [16]?
Committed Information Rate (CIR) in bps [64000]?
Committed Burst Size (Bc) in bits [64000]?
Excess Burst Size (Be) in bits [0]?
Assign Circuit name []?
Is circuit required for interface operation [N]?
Does the circuit belong to a required PVC group [N]?
What is the group name []?
Do you want to have data compression performed [Y]?
```

Circuit Number

Indicates the circuit number for this PVC.

Valid Values: 16 to 1007.

Note: If you are configuring CLLM to help control congestion, you cannot configure 1007 as a PVC.

Committed Information Rate

Indicates the committed information rate (CIR). The CIR can be either 0 or a value in the range 300 bps to 2 048 000 bps. For more information, see “Committed Information Rate (CIR)” on page 31-9. The default is the value of the default CIR configured for the interface.

Committed Burst Size

The maximum amount of data in bits that the network agrees to deliver during a measurement interval equal to committed burst (Bc) size / CIR seconds. The range is 300 to 2048000 bits. The default value is value of the default committed burst configured for the interface.

Note: If CIR is configured as 0 then the committed burst size is set to 0 and you are not prompted for a value. For additional information, see “Committed Burst (Bc) Size” on page 31-10.

Excess Burst Size
The maximum amount of uncommitted data in bits in excess of committed burst size that the network attempts to deliver during a measurement interval equal to (Committed Burst Size/CIR) seconds. Range is 0 to 2048000 bits. The default value is the value configured for excess burst size for the interface. For additional information, see “Excess Burst (Be) Size” on page 31-10.

Assign Circuit Name
Indicates the ASCII string that is assigned to describe the circuit. The default is unassigned.

Is the circuit required for operation
Specify Y or N to indicate whether the circuit is required for interface operation.

Does the circuit belong to a required PVC group
This prompt is displayed only for circuits that are required. Specify Y or N to indicate whether the circuit should belong to a required PVC group.

What is the group name
Enables you to specify the name of the required PVC group when the PVC is defined as belonging to a required group. Enter a question mark (?) for a list of currently defined groups.

Do you want to have compression performed
Enables you to specify whether or not the circuit will compress data packets. This question appears only if compression is enabled on the interface.

Note: If you enable compression on a PVC and exceed the interface’s compression PVC limit, you will get a message. Compression will be performed on the circuit, if possible – that is, the active compression limit has not been exceeded when the circuit becomes active.

protocol-address
This command adds statically configured destination protocol (protocol-name) addresses to the Frame Relay interface. Statically configured destination protocol addresses are useful if neither Inverse ARP nor ARP is an option, or for other reasons such as security. Adding protocol name and address mappings (static ARP) is less efficient than Inverse ARP or ARP.

- Inverse ARP is the preferred, efficient method because of dynamic address mapping with no broadcasts.
- ARP is recommended if Inverse ARP is not an option. It is less efficient than Inverse ARP because it uses address broadcast and mappings are relearned at regular intervals.

This parameter prompts you for different information depending on the type of protocol that you are adding.

Example: **`add protocol-address`**

```
Protocol name or number [0]?
```

IP protocol:

```
IP Address [0.0.0.0]?
Circuit Number [16]?
```

IPX protocol:

```
Host Number (in hex)[]?
Circuit Number [16]?
```

AppleTalk Phase 2 protocol:

```
Network Number (1-65279) []?
Node Number (1-253) []?
Circuit Number [16]?
```

DN protocol:

```
Node address [0.0]?
Circuit Number [16]?
```

Protocol name or number
Defines the name or number of the protocol that you are adding. If you should specify an unsupported protocol, the system will prompt you with the error message:

```
Unknown protocol name, try again
```

For example, you may have erroneously specified one of the following:

```
Prot#  Name
0      IP
4      DN
7      IPX
22     AP2
```

To see a list of supported protocol types, type `?` at the `Protocol name or number [IP]?` prompt.

IP Address
Defines the 32-bit Internet address in dotted-decimal notation of the remote IP host.

Host Number
Defines the 48-bit IPX node address of the remote IPX host.

Network Number
Defines the AppleTalk Phase 2 network number of the remote AppleTalk host.

Node Number
Defines the node number of the interface attached to the remote AppleTalk host.

Node address
Defines the DECnet node address of the remote DECnet host. Configure the node address in the format *x.y*, where *x* is a 6-bit area address and *y* is a 10-bit node number.

Circuit Number
Defines the PVC in the range 16 to 1007 that this protocol is to run over.

pvc-group

Adds a Required PVC group name.

Example: **add pvc-group**

```
PVC group name []? group1
```

Change

Use the **change permanent-virtual-circuit** command to change any previous PVCs that were added with the **add permanent-virtual-circuit** command.

Syntax: change permanent-virtual-circuit . . .

Example: **change permanent-virtual-circuit**

```
Circuit Number [16]?
Committed Information Rate in bps [64000]?
Committed Burst Size (Bc) in bits [64000]?
Excess Burst Size (Be) in bits [0]?
Assign Circuit Name: []?
Is the circuit required for interface operation [N]?
Does the circuit belong to a required group [N]?
What is the group name []?
Do you want to have data compression performed []?
```

Circuit Number

Indicates the circuit number for this PVC.

Valid Values: 16 to 1007.

Note: If you are configuring CLLM to help control congestion, you cannot configure 1007 as a PVC.

Committed Information Rate

Indicates the committed information rate (CIR). The CIR can be either 0 or a value in the range 300 bps to 2048000 bps. The default for an interface is 64000 bps, but the default for an individual circuit is the value configured with the **set cir-defaults** command.

Committed Burst Size

The maximum amount of data in bits that the network agrees to deliver during a measurement interval equal to (Committed Burst Size/CIR) seconds. If the CIR is configured as 0, the committed burst size is also set to 0. Otherwise, the range of valid values is 300 to 2048000 bits. The default for an interface is 64000 bits, but the default for an individual circuit is the value configured with the **set cir-defaults** command.

Excess Burst Size

The maximum amount of uncommitted data in bits in excess of Committed Burst Size that the network attempts to deliver during a measurement interval equal to (Committed Burst Size/CIR) seconds. Range is 0 to 2048000 bits. The default for an interface is 64000 bps, but the default for an individual circuit is the value configured with the **set cir-defaults** command.

Assign circuit Name

Indicates the ASCII character string designation for the circuit that you want to change.

Is the circuit required for operation

Specify Y or N to indicate whether the circuit is required for interface operation.

Does the circuit belong to a required PVC group

This prompt is only displayed for circuits that are required. Specify Y or N to indicate whether the circuit should belong to a required PVC group.

What is the group name

Enables you to specify the name of the required PVC group when the PVC is defined as belonging to a required group. Enter a question mark (?) for a list of currently defined groups.

Do you want to have data compression performed

Enables you to specifies whether or not the circuit will compress data packets. This question appears only if compression is enabled on the interface.

Note: If you enable compression on a PVC and exceed the interface's compression PVC limit, you will get a message. Compression will be performed on the circuit, if possible – that is, the active compression limit has not been exceeded when the circuit becomes active.

Disable

Use the **disable** command to disable those features previously enabled using the **enable** command.

Syntax: disable
- cir-monitor
- cllm
- compression
- congestion-monitor
- dn-length-field
- lmi
- lower-dtr
- multicast-emulation
- no-pvc
- notify-fecn-source
- orphan-circuits
- protocol-broadcast
- throttle-transmit-on-fecn

cir-monitor

Disabling this feature allows the circuit's information rate to exceed the maximum information rate that is calculated using the parameters configured with the **add permanent-virtual-circuit** command. The default setting for this feature is disabled. See "Circuit Congestion" on page 31-12 for more information.

Example: `disable cir-monitor`

cllm

Disables the device from *throttling down* in response to a CLLM message. The default is disabled. See "Circuit Congestion" on page 31-12.)

Example: `disable cllm`

compression

Disables compression on the interface. Compression will not be performed for any PVC.

Example: `disable compression`

congestion-monitor

Disables the congestion monitoring feature. Disabling this feature prevents a circuit's information rate from varying in response to congestion between the minimum information rate and the line speed. See "Circuit Congestion" on page 31-12 for more information. The default setting for this feature is enabled.

Example: `disable congestion-monitor`

dn-length-field

Prevents inter-operation with implementations of DECnet Phase IV over Frame Relay that require a length field to precede DECnet packets in Frame Relay frames, but allows inter-operation with DECnet Phase IV Frame Relay software that does not use a length field before the DECnet packet. Disabling dn-length-field causes Frame Relay not to insert a length field into transmitted frames containing DECnet packets and not to attempt to remove the length field from received frames containing DECnet packets.

Note: This option is presented as a configuration option only

Example: `disable dn-length-field`

lmi

Note: Disabling this parameter allows for normal operation or end-to-end Frame Relay testing in the absence of a real network or management interface. With end-to-end Frame Relay testing, it is necessary to add like PVCs (the same PVC number, such as 16 and 16) on both ends of the link.

Example: `disable lmi`

lower-dtr

This parameter determines how the data terminal ready (DTR) signal is handled for leased serial-line interfaces on the router. It is not supported on Frame Relay dial circuit interfaces. See the **enable lower-dtr** command for a more complete description of the lower-dtr parameter.

The following cable types are supported:

EIA 232 (RS-232)
V.35
V.36

The default setting is **disable lower-dtr**.

Example: `disable lower-dtr`

multicast-emulation

Disables multicast emulation on each active PVC. The default setting for this feature is enabled. If you disable this feature, you are required to add protocol static address maps.

Some protocols, such as IPX RIP, will not function on the Frame Relay interface if multicast-emulation is disabled. The protocol-broadcast feature also requires multicast-emulation in order to function properly. For more information, see "Multicast Emulation and Protocol Broadcast" on page 31-8.

Example: `disable multicast-emulation`

no-pvc
Controls whether the interface is considered active or inactive. If no-pvc is disabled, the presence of active PVCs on the interface does not affect whether the Frame Relay interface is considered active or inactive.

notify-fecn-source
Disables setting a BECN bit on the first packet destined to a device from which the router received a packet with the FECN bit set. See "Circuit Congestion" on page 31-12 for more information.

Example: `disable notify-fecn-source`

orphan-circuits
Prohibits the use of all non-configured orphan circuits at the interface. The default setting for orphan circuits is enabled. Disabling orphan circuits adds a measure of security to your network by preventing unauthorized entry from a non-configured circuit. However, if you disable orphan circuits, you are required to add PVCs that will be used on the interface.

Example: `disable orphan-circuits`

protocol-broadcast
Prohibits protocols such as IP RIP from functioning over the Frame Relay interface. For more information, see "Multicast Emulation and Protocol Broadcast" on page 31-8. The default setting for this feature is enabled.

Example: `disable protocol-broadcast`

throttle-transmit-on-fecn
Prohibits the device from *throttling down* the transmission of packets in response to a packet with a FECN bit set on. The default is disabled. See "Circuit Congestion" on page 31-12 for more information.

Example: `disable throttle-transmit-on-fecn`

Enable

Use the **enable** command to enable Frame Relay features.

Syntax: enable
- cir-monitor
- cllm
- compression
- congestion-monitor
- dn-length-field
- lmi
- lower-dtr
- multicast-emulation
- notify-fecn-source
- no-pvc
- orphan-circuits
- protocol-broadcast
- throttle-transmit-on-fecn

cir-monitor
Enables the circuit monitoring feature. The circuit monitoring feature ensures that the circuit's information rate varies between the minimum information rate and the maximum information rate, calculated using the parameters configured

with the **add permanent-virtual-circuit** command or the **change permanent-virtual-circuit** command

Note: The circuit monitoring feature overrides the congestion monitoring feature if there is a conflict when both are enabled. The default setting for this feature is disabled.

For additional information on CIR monitoring, see "CIR Monitoring" on page 31-12.

Note: To maximize throughput for circuits running data compression, you should not enable CIR monitoring on the same interface on which you have enabled compression. Because the device uses the uncompressed size of frames to determine if the VIR of a PVC is being exceeded and compressed frames will require less bandwidth, the CIR of a PVC will be under-utilized if the device strictly monitors and does not exceed the configured CIR. Instead, congestion monitoring can be used to allow the device to react to congestion indications sent by the FR network to avoid frame loss.

Example: `enable cir-monitor`

cllm

Enables the device to *throttle down* in response to a CLLM message. Contact your FR network provider to see whether this support is available. See "Circuit Congestion" on page 31-12 for more information.

Example: `enable cllm`

compression

Enables compression on the interface. All compression-capable PVCs on the interface can compress data packets, provided that contexts are available and the active compression PVC limit has not been exceeded. (See Chapter 19, "The Data Compression Subsystem" on page 19-1 for details.)

Note: To maximize throughput for circuits running data compression, you should not enable CIR monitoring on the same interface on which you have enabled compression. Because the device uses the uncompressed size of frames to determine if the VIR of a PVC is being exceeded and compressed frames will require less bandwidth, the CIR of a PVC will be under-utilized if the device strictly monitors and does not exceed the configured CIR. Instead, congestion monitoring can be used to allow the device to react to congestion indications sent by the FR network to avoid frame loss.

Example: `enable compression`

congestion-monitor

Enables the congestion monitoring feature. This feature allows a circuit's information rate to vary in response to congestion between the minimum information rate and the line speed.

Note: The circuit monitoring feature overrides the congestion monitoring feature if there is a conflict when both are enabled. The default setting for this feature is enabled.

For additional information on congestion monitoring, see "Congestion Monitoring" on page 31-12.

Example: `enable congestion-monitor`

dn-length-field

Supports inter-operation with implementations of DECnet Phase IV over Frame Relay that require a length field to precede DECnet packets in Frame Relay frames. Enabling dn-length-field causes Frame Relay to insert a length field into transmitted frames containing DECnet packets and to remove the length field from received frames containing DECnet packets. This option is disabled by default. By default, Frame Relay will neither insert nor attempt to remove the length field.

Note: This option is presented as a configuration option only when the router software contains the DECnet Phase IV protocol.

Example: `enable dn-length-field`

lmi

Enables management activity.

After issuing the **enable lmi** command, use the **set lmi-type** command to select the management mode for your Frame Relay interface. See “Enabling Frame Relay Management” on page 31-16. The system defaults to ANSI T1.617 Annex D management.

Use the **enable lmi** command to resume LMI management if you have previously disabled Frame Relay management.

Example: `enable lmi`

lower-dtr

This parameter determines how the data terminal ready (DTR) signal is handled for leased serial-line interfaces that are disabled. It is not supported on Frame Relay dial circuit interfaces. If this parameter is set to “disabled” (the default), the DTR signal will remain raised when the interface is disabled.

When lower-dtr is enabled, DTR will be lowered when the interface is disabled. This behavior may be desirable in situations where the interface has been configured as an alternate link for WAN Reroute and the interface is connected to a dial-out modem which maintains its dial connection based on the state of the DTR signal.

If this feature is enabled and the interface is disabled, the DTR signal is low and the modem keeps the dial connection down. When the interface is enabled, due to a WAN Reroute backup scenario, DTR is raised and the modem dials a stored number to the backup site. When the primary interface is restored, the alternate interface is disabled, DTR is lowered, and the modem hangs up the dial connection.

The following cable types are supported:

EIA 232 (RS-232)
V.35
V.36

The default setting is **disable lower-dtr**.

Example: `enable lower-dtr`

multicast-emulation

Enables multicast emulation. This allows a multicast/broadcast frame to be transmitted on each active PVC. Protocols such as ARP, IPX RIP, and IP RIP require multicast emulation to be enabled to function correctly over a Frame

Relay interface. For more information, see “Multicast Emulation and Protocol Broadcast” on page 31-8. The default for this parameter is enabled.

Example: `enable multicast-emulation`

no-pvc
Controls whether the interface is considered active or inactive. When this feature is enabled, the Frame Relay interface becomes inactive when there are no active PVCs on the interface. If at least one PVC is active, the Frame Relay interface becomes active when a successful LMI exchange occurs between the router and the FR switch.

notify-fecn-source
Enables setting a BECN bit on the first packet destined to a device from which the router received a packet with the FECN bit set. Use this parameter to enhance the congestion control mechanisms of the device in a network whether the FR switches do not themselves set BECN but set FECN. See “Circuit Congestion” on page 31-12 for more information.

Example: `enable notify-fecn-source`

orphan-circuits
Enables the use of all non-configured orphan circuits. The default for this feature is enabled. See “Orphan Circuit CIR” on page 31-10 for information about the default CIR values.

Example: `enable orphan-circuits`

protocol-broadcast
Allows protocols such as IP RIP to function correctly over the Frame Relay interface. The multicast emulation feature must be enabled for the protocol-broadcast feature to function correctly. The default setting for this feature is enabled.

Example: `enable protocol-broadcast`

throttle-transmit-on-fecn
Enables the device to *throttle down* the transmission of packets in response to a packet with a FECN bit set on. Use this parameter to minimize overall FR network congestion whenever a congestion indication is received. It causes the device to react to a FECN in the same way that it reacts to a BECN.

Example: `enable throttle-transmit-on-fecn`

List

Use the **list** command to display currently configured management and PVC information.

Syntax:; list
- all
- hdlc
- lmi
- permanent-virtual-circuits
- protocol-address
- pvc-groups

all
Displays the Frame Relay configuration.

See **list hdlc** and **list lmi** for descriptions of the parameters.

Example: `list all`

```
                              Frame Relay HDLC Configuration


 Interface MTU in bytes = 2048
 Encoding               = NRZ
 Idle state             = Flag
 Clocking               = External
 Cable type             = V.35 DTE
 Line speed (bps)       = 64000
 Transmit delay         = 0
 Lower DTR              = Enabled


                              Frame Relay Configuration


 LMI enabled            =   Yes  LMI DLCI                       = 1023
 LMI type               =  REV1  LMI Orphans OK                 =  Yes
 CLLM enabled           =   Yes  Timer Ty seconds               =   10

 Protocol broadcast     =   Yes  Congestion monitoring          =  Yes
 Emulate multicast      =   Yes  CIR monitoring                 =   No
 Notify FECN Source     =   Yes  Throttle Transmit on FECN      =  Yes

 Data compression       =   Yes  Orphan compression               =   No
 Compression PVC limit  =    10  Number of compression PVCs     =    3 [1]

 PVCs P1 allowed        =    64  Interface down if no PVCs      =   No
 Timer T1 seconds       =    10  Counter N1 increments          =    6
 LMI N2 error threshold =     3  LMI N3 error threshold window =    4
 MIR % of CIR           =    25  IR % Increment                 =   12
 IR % Decrement         =    25  DECnet length field            =   No

MAXIMUM PVCs allowable =      64
Total PVCs configured  =       7

            Circuit              Circuit   Circuit   CIR     Burst  Excess
            Name                 Number     Type    in bps   Size   Burst
-------------------------------- ------- ---------- ------- ------- -------

 cir16                             16   @#Permanent   64000   64000
 cir244                           244    #Permanent   64000   64000     0
 cir33                             33    #Permanent   64000   64000     0
 cir1005                         1005    #Permanent   64000   64000     0
 cir55                             55    @Permanent   64000   64000
 cir22                             22    #Permanent   64000   64000     0
 cir66                             66   @*Permanent   64000   64000

* = circuit is required
# = circuit is required and belongs to a required PVC group
@ = circuit is data compression capable

Required PVC group = group1

      Circuit #  16
      Circuit #  244
      Circuit #  22


Required PVC group = group2

      Circuit #  33
      Circuit #  1005


No address translations configured
```

Note: [1] – This line only appears when data compression is on (yes).

hdlc

Displays the Frame Relay High-Level Data Link Control (HDLC) configuration.

Example: **`list hdlc`**

```
                    Frame Relay HDLC Configuration


Interface MTU in bytes = 2048
Encoding               = NRZ
Idle state             = Flag
Clocking               = External
Cable type             = V.35 DTE
Line speed (bps)       = 64000
Transmit delay         = 0
Lower DTR              = Enabled
```

Encoding The transmission encoding scheme for the serial interface. Encoding is NRZ (non-return to zero) or NRZI (non-return to zero inverted).

Idle The data link idle state: flag or mark.

Clocking The type of clocking: internal or external.

Cable type The serial adapter cable type: RS-232, V.35, V.36, or X.21.

Line Speed (bps)
Indicates the physical data rate for the Frame Relay interface.

Interface MTU in bytes
Indicates the maximum transmission unit (amount of user data per frame) that can be transmitted or received over the network at any given time.

Transmit delay
Indicates the number of flag bytes sent between frames.

Lower DTR Indicates whether the router will drop the DTR signal when a WAN Reroute alternate link is no longer needed. Dropping the DTR signal causes the modem to terminate the leased-line connection for the alternate link. Lower DTR does not appear when the cable type is X.21.

Note: For a FR dial circuit interface, only the Interface MTU is displayed.

lmi

Displays logical management and related configuration information about the Frame Relay interface.

Example: **`list lmi`**

```
                                Frame Relay Configuration

LMI enabled            =  Yes   LMI DLCI                        =    0
LMI type               =  ANSI  LMI Orphans OK                  =  Yes
CLLM enabled           =   Yes  Timer Ty seconds    = 10

Protocol broadcast     =   Yes  Congestion monitoring           =  Yes
Emulate multicast      =   Yes  CIR monitoring                  =   No
Notify FECN Source     =   Yes  Throttle Transmit on FECN       =  Yes

Data compression       =  Yes   Orphan compression             =   No
Compression PVC limit  =   10   Number of compression PVCs     =    5 1

PVCs P1 allowed        =    64  Interface down in no PVCs       =   No
Timer T1 seconds       =    10  Counter N1 increments           =    6
LMI N2 error threshold =     3  LMI N3 error threshold window =    4
MIR % of CIR           =    25  IR % Increment                  =   25
IR % Decrement         =    25  DECnet length field            =   No
Default CIR            = 64000  Default Burst Size              = 64000
Default Excess Burst   =     0
```

Note: **1** – This line appears only when data compression is on (yes).

LMI enabled
Indicates whether the management features are enabled on the Frame Relay interface, yes or no.

LMI DLCI
Indicates the management circuit number. This number reflects the LMI type: 0 for ANSI and ITU-T/CCITT and 1023 for REV1.

LMI Type
Indicates the LMI type: REV1, ANSI, or CCITT.

LMI Orphans OK
Indicates if non-configured circuits are available for use, yes or no.

CLLM Enabled
Indicates whether CLLM is enabled on the Frame Relay interface.

Timer Ty seconds
Indicates the amount of time that must elapse without the device receiving any CLLM messages or BECNs before the device considers a congestion condition cleared and gradually return the PVC to its configured transmission rate.

Protocol Broadcast
Indicates whether protocols such as IP RIP can function over the Frame Relay interface, yes or no.

Emulate multicast
Indicates whether the multicast emulation feature is enabled on each active PVC, yes or no.

Congestion Monitoring
Indicates whether the congestion monitoring feature that responds to network congestion is enabled, yes or no.

CIR monitoring
Indicates whether the circuit monitoring feature that enforces the transmission rate is enabled, yes or no.

Notify FECN Source
Indicates whether this device sets a BECN bit on the first packet destined to a device from which the router received a packet with the FECN bit set.

Throttle Transmit on FECN
Indicates whether the device will *throttle down* the transmission of packets in response to a packet with a FECN bit set on.

Data compression
Indicates whether this interface has data compression enabled.

Orphan compression
Indicates whether orphan circuits on this interface will have data compression enabled.

Note: Enabling compression on orphan circuits will decrease the number of available compression contexts available for the native PVCs on the device.

Compression PVC limit
Indicates the maximum number of PVCs that can participate in data compression.

Number of compression PVCs
Indicates the current number of PVCs compressing data.

PVCs P1 allowed
Indicates the number of allowable PVCs for use with this interface.

Timer T1 seconds
Indicates the frequency with which the Frame Relay interface performs a sequence number exchange with the Frame Relay switch LMI entity.

Counter N1 increments
Indicates the number of T1 timer intervals which must expire before a complete PVC LMI status enquiry is made.

LMI N2 error threshold
Indicates the number of management event errors occurring within the N3 window that will cause a reset of the Frame Relay interface.

LMI N3 error threshold window
Indicates the number of monitored management events used to measure the N2 error threshold.

MIR % of CIR
Minimum IR, expressed as a percentage of CIR.

IR % Increment
Percentage by which the router increments the IR each time it receives a frame without BECN until it reaches the maximum IR.

IR % Decrement
Percentage by which the router decrements the IR each time it receives a frame that contains BECN until it reaches the minimum IR.

Default CIR
The committed information rate, in bits per second, used as the default for PVCs on this interface.

Default Burst Size
The committed burst size, in bits, used as the default for PVCs on this interface.

Default Excess Burst Size
The excess burst size, in bits, used as the default for PVCs on this interface.

permanent-virtual-circuits
Displays all the configured PVCs on the Frame Relay interface.

Example: `list permanent-virtual-circuit`

```
FR Config>li perm

  Maximum PVCs allowable =      64
  Total PVCs configured  =       7

             Circuit              Circuit   Circuit   CIR     Burst  Excess
              Name                Number     Type    in bps   Size   Burst
 -------------------------------- ------- ---------- ------- ------- -------
 cir16                              16   @#Permanent   64000   64000       0
 cir244                             244   #Permanent   64000   64000       0
 cir33                              33    #Permanent   64000   64000       0
 cir1005                            1005  #Permanent   64000   64000       0
 cir55                              55    #Permanent   64000   64000       0
 cir22                              22    @Permanent   64000   64000       0
 cir66                              66   @*Permanent   64000   64000       0

* = circuit is required
# = circuit is required and belongs to a Required PVC group
@ = circuit is data compression capable
```

Maximum PVCs allowable
Indicates the number of PVCs that can exist for this interface. This number includes any PVCs that you added with the **add permanent-virtual-circuit** command and dynamically learned through the management interface.

Total PVCs configured
Indicates the total number of currently configured PVCs for this interface.

Circuit Name
Indicates the ASCII designation of the configured PVC.

Circuit Number
Indicates the number of a currently configured PVC.

Circuit Type
Indicates the type of virtual circuit currently configured. This release of Frame Relay only supports permanent virtual circuits.

Committed Information Rate
Indicates the information rate at which the network agrees to transfer data under normal conditions.

Committed Burst Size
The maximum amount of data in bits that the network agrees to deliver during a measurement interval equal to (Committed Burst Size/CIR) seconds.

Excess Burst Size
The maximum amount of uncommitted data in bits in excess of Committed Burst Size that the network attempts to deliver during a measurement interval equal to (Committed Burst Size/CIR) seconds.

pvc-groups

Displays all the Required PVC groups on the Frame Relay interface.

Example: **`list pvc-groups`**

```
Required PVC group = group1

      Circuit # 16
```

protocol-addresses

Displays all the statically configured protocol addresses of circuit mappings at the Frame Relay interface.

Example: **`list protocol-addresses`**

```
                Frame Relay Protocol Address Translations

 Protocol Type              Protocol Address      Circuit Number
 -------------              ----------------      --------------
      IP                     125.2.29.4                  21
      IPX                    000000004503                16
```

Protocol Type Displays the name of the protocol running over the interface.

Protocol Address Displays the protocol address of the device at the other end of the circuit.

Circuit Number Displays the PVC that is handling the protocol.

LLC

Use the **LLC** command to access the LLC configuration environment. See “LLC Configuration Commands” on page 24-1 for an explanation of each of these commands.

Note: The **LLC** command is supported only if APPN is in the software load.

Syntax: llc

Example: **`llc`**

```
LLC config>
```

Remove

Use the **remove** command to delete any PVC, Required PVC group, or protocol-address previously added using the **add** command.

Syntax: remove permanent-virtual-circuit . . .
protocol-address
pvc-group

permanent-virtual-circuit *pvc#*

Deletes any configured PVC in the range 16 to 1007.

Note: When you delete a PVC that is running compression, the interface decreases the count of active compression PVCs. If this action brings the count of compression PVCs below the limit, you will receive a message to that effect.

Example: **`remove permanent-virtual-circuit 20`**

protocol-address

Deletes any configured protocol addresses (static ARP entries). This parameter prompts you for different information depending on the type of protocol that you are adding.

Example: **remove protocol-address**

```
Protocol name or number [IP]?
```

pvc-group

Deletes any configured PVC group by name. The group is removed only if it has no member circuits.

Example: **remove pvc-group**

```
PVC group name [IP]?
```

IP protocol:

```
IP Address [0.0.0.0]?
Circuit Number [16]?
```

IPX protocol:

```
Host Number (in hex)[]?
Circuit Number [16]?
```

AppleTalk Phase 2 protocol:

```
Network Number (1-65279) []?
Node Number (1-253) []?
Circuit Number [16]?
```

DN protocol:

```
Node address [0.0]?
Circuit Number [16]?
```

Protocol name or number

Defines the name or number of the protocol that you are deleting. If you try to delete an unsupported protocol the system will display the error message:

```
Unknown protocol name, try again
```

To see a list of supported protocols, type `?` at the `Protocol name or number [IP]?` prompt.

IP Address

Defines the 32-bit internet address of the remote IP host in dotted-decimal notation.

Host Number

Defines the 48-bit node address of the remote IPX host.

Network Number

Defines the AppleTalk Phase 2 network number.

Node Number

Defines the node number of the interface attached to the remote AppleTalk host.

Node address

Defines the DECnet node address of the remote DECnet host. Configure the node address in the format *x*,*y*, where *x* is a 6-bit area address and *y* is a 10-bit node number.

Circuit Number

Defines the PVC in the range 16 to 1007 that the protocol runs over.

Set

Use the **set** command to configure the interface to run the Frame Relay protocol.

Set Command Considerations

Two parameters, the n2-parameter and the n3-parameter, require further explanation before you configure them. The n2-parameter sets the error threshold for management events, and the n3-parameter sets the number of events that are monitored in the event window. If the number of management errors in the event window equals n2, the Frame Relay interface resets. For example:

set n3-parameter 4
set n2-parameter 3

You now have a window size of 4 (n3 = 4) and an error threshold of 3 (n2 = 3). That means the system is monitoring 4 management events and checking to determine if any of those are in error. If the number of events in error equals 3 (the n2 parameter), the Frame Relay interface is reset and the status of the network is considered *network down*.

For the status of the network to be considered *network up*, the number of events in error within the window must be less than n2 prior to any change in status.

Syntax: set
- cable*
- cir-defaults
- clocking*
- encoding*
- frame-size
- idle . . .*
- ir-adjustment . . .
- line-speed*
- lmi-type n1-parameter
- n2-parameter
- n3-parameter
- p1-parameter
- t1-parameter
- transmit-delay . . .*
- ty-parameter

*** Note:** The commands with an * following them are not available for FR dial circuit interfaces.

cable *physical-interface-link-type data-connection-type*
Sets the cable type for the network physical link.

A DTE cable is used when you are attaching the router to some type of DCE device (for example, a modem or a DSU/CSU). A DCE cable is used when the router is acting as the DCE and providing the clocking for direct attachment.

The available options are:

Physical Interface Link Type	Data Connection Type
EIA 232 (RS-232)	DTE, DCE
V35	DTE, DCE
V36	DTE
X21	DTE, DCE

Example: **`set cable rs-232 dte`**

cir-defaults

Sets the default values for the circuit congestion parameters. The parameters are:

cir Sets the default value of *cir* to the value provided by a Frame Relay network provider.

Valid Values: 0 or 300 to 204 800 bps

Default Value: 64 000

bc Sets the default value of *bc* to the value provided by a Frame Relay network provider.

Valid Values: See “Committed Burst (Bc) Size” on page 31-10

Default Value: 64 000

be Sets the default value of *be* to the value provided by a Frame Relay network provider.

Valid Values: See “Excess Burst (Be) Size” on page 31-10

Default Value: 0

Example:

```
FR 6 config> set cir-default
Default Committed Information Rate (CIR) in bps [64000]? 48000
Default Committed Burst Size (Bc) in bits [64000]? 40000
Default Excess Burst Size (Be) in bits [0]? 52000
```

clocking *external* or *internal*

To connect to a modem or DSU, configure clocking as external. To connect directly to another DTE device, use a DCE cable and set the clocking to internal. For internal clocking, you must enter the **set line-speed** command to configure a clock speed between 2400 and 2048000 bps.

For external clocking the maximum line speed is 6 312 000 bps.

Example: **`set clocking internal`**

encoding *NRZ* or *NRZI*

Sets the HDLC transmission encoding scheme as NRZ (non-return to zero) or NRZI (non-return to zero inverted). Most configurations use NRZ, which is the default.

Example: **`set encoding nrz`**

frame-size *#*

Sets the size of the network layer portion of frames transmitted and received on the data link. This size includes the two bytes containing

the DLCI and user data. See Figure 31-4 on page 31-5 for more information. Values are 5 to 8190. The default is 2048.

Example: `set frame-size 2000`

idle *flag* or *mark*
Sets the transmit idle state for HDLC framing. The default value is **flag**, which provides continuous flags (7E hex) between frames. The mark option puts the line in a marking state (OFF, 1) between frames. Mark idle causes the transmit LED to be dark between frames. Flag idle partially lights the transmit LED between frames.

Example: `set idle flag`

ir-adjustment *increment-% decrement-% minimum-IR*
Sets the minimum information rate (IR) and the percentages for incrementing and decrementing the IR in response to network congestion.

The minimum IR, expressed as a percentage of CIR, is the lower limit of the information rate. The minimum percentage is 1 and the maximum percentage is 100. The default is 25.

When network congestion clears, the information rate is gradually incremented by the IR adjustment increment percentage until the maximum information rate is reached. The minimum percentage is 1 and the maximum percentage is 100. The default is 12.

When network congestion occurs, the information rate is decremented by the IR adjustment decrement percentage each time a frame containing BECN is received until the minimum information rate is reached. The minimum percentage is 1, and the maximum percentage is 100. The default is 25.

Example: `set ir-adjustment`

```
IR adjustment % increment [1₂]?
IR adjustment % decrement [25]?
Minimum IR as % of CIR [25]?
```

line-speed *rate*
For internal clocking, this command specifies the speed of the transmit and receive clock lines. The range is 2400 to 2 048 000 bps.

For external clocking, this command does not affect the hardware (in other words, the actual speed of the line) but it sets the speed some protocols, such as IPX, use to determine routing cost parameters. Congestion monitoring also uses the configured line speed to determine the maximum information rate. Therefore, it is recommended that you set the speed to match the actual line speed. If the speed is not configured, the protocols and congestion monitoring assume a speed of 1 000 000 bps.

Notes:

1. When using external clocking, the maximum line speed is 6 312 000.
2. When using internal clocking, the maximum line speed is 2 048 000.

Example: `set line-speed 64000`

lmi-type *rev1* or *ansi* or *ccitt*

Sets the management type for the interface. See "Enabling Frame Relay Management" on page 31-16 for details on setting Frame Relay management. The default is type *ansi* enabled.

Table 31-4. Frame Relay Management Options

Command	Management Type	Description
set	lmi-type rev1	Conforms to LMI Revision 1, (Stratacom's Frame Relay Interface Specification)
set	lmi-type ansi	Conforms to ANSI T1.617 ISDN-DSS1-Signalling Specification for Frame Relay Bearer Service (known as Annex D)
set	lmi-type ccitt	Conforms to Annex A of ITU-T/CCITT Recommendation Q.933 - DSS1 Signalling Specification for Frame Mode Basic Call Control.

Example: **`set lmi-type rev1`**
or
`set lmi-type ansi`
or
`set lmi-type ccitt`

n1-parameter *count*

Configures the number of T1 timer intervals which must expire before a complete PVC status enquiry is made. *Count* is the interval in the range 1 to 255. The default is 6.

Example: **`set n1-parameter`**

```
Parameter N1 [6]?
```

n2-parameter *max#*

Configures the number of errors that can occur in the management event window monitored by the n3-parameter before the Frame Relay interface resets. Max# is a number in the range 1 to 10. The default is 3. This parameter must be less than or equal to the n3-parameter or you will receive an error message.

Example: **`set n2-parameter`**

```
Parameter N2 [3]?
```

n3-parameter *max#*

Configures the number of monitored management events for measuring the n2-parameter. Max# is a number in the range 1 to 10. The default is 4.

Example: **`set n3-parameter`**

```
Parameter N3 [4]?
```

p1-parameter *max#*

Configures the maximum number of PVCs supported by the Frame Relay interface. This includes active, inactive, removed, and congested PVCs. Max# is a number in the range 0 to 992. The default is 64. 0 (zero) implies that the interface supports no PVCs.

Example: **set p1-parameter**

```
Parameter P1 [64]?
```

t1-parameter *time*

Configures the interval (in seconds) between sequence number exchanges with Frame Relay management. The management's T2 timer is the allowable interval for an end station to request a sequence number exchange with the manager. The T1 interval must be less than the T2 interval of the network. *Time* is a number in the range 5 to 30. The default is 10.

Example: **set t1-parameter**

```
Parameter T1  [10]?
```

transmit-delay *#*

Allows the insertion of a delay between transmitted packets. The purpose of this command is to slow the serial line so that it is compatible with older, slower serial devices at the other end. It can also prevent the loss of serial line hello packets between the lines. *#* is between 0 and 15 extra flags. The default is zero (0). Setting this parameter provides 0 to 15 extra flags between transmit frames. Table 31-5 lists the units and range values for serial interfaces.

Table 31-5. Transmit Delay Units and Range for the 2210 Serial Interface

Unit	Minimum	Maximum
Extra Flags	0	15

Example: **set transmit-delay 15**

ty-parameter *time*

Configures the interval after which the device considers an existing congestion condition indicated by the receipt of a CLLM message to be cleared. If the device receives a CLLM message before the timer expires, the device resets this timer.

Valid Values: 5 to 30 seconds.

Default Value: 11 seconds.

Example: **set ty-parameter**

```
Parameter Ty  [11]? 15
```

Exit

Use the **exit** command to return to the `Config>` prompt.

Syntax: exit

Example: **exit**

Chapter 32. Monitoring Frame Relay Interfaces

This chapter describes the Frame Relay console commands and includes the following sections:

Note: For information on monitoring bandwidth reservation over Frame Relay, refer to Chapter 11, "Monitoring Bandwidth Reservation" on page 11-1.

Displaying the Frame Relay Console Prompt

To access the Frame Relay console commands and to monitor Frame Relay on your router, perform the following steps:

1. At the OPCON prompt (*), type **talk 5**.
2. At the GWCON prompt (+), enter the **interface** command to see a list of interfaces configured on the router.
3. Enter the **network** command followed by the network number of the frame relay interface. For example:

```
+ net 2
Frame Relay Console
FR 2 >
```

Frame Relay Console Commands

This section summarizes and then explains the Frame Relay Console commands. Use these commands to gather information from the database. Table 32-1 shows the commands.

Table 32-1. Frame Relay Console Commands Summary

Command	Function
? (Help)	Displays all the Frame Relay console commands (clear and list) or any options associated with those commands.
Clear	Clears statistical information on the Frame Relay interface.
Disable	Disables CIR monitoring and congestion monitoring on the Frame Relay interface.
Enable	Enables CIR monitoring and congestion monitoring on the Frame Relay interface.
List	Displays statistics specific to the data-link layer and Frame Relay management.
LLC	Displays the LLC console monitoring prompt.
Set	Sets CIR, Committed Burst Size, and Excess Burst Size for a Frame Relay PVC.
Exit	Exits the Frame Relay console process.

Note: In this section, the terms *circuit number* and *PVC* are equivalent to the term *data link circuit identifier (DLCI)*.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
CLEAR
DISABLE
ENABLE
LIST
SET
EXIT
```

Clear

Use the **clear** command to remove all statistics on the Frame Relay interface.

Note: Statistics can also be cleared by using the OPCON **clear** command.

Syntax: clear

Example: **clear**

Disable

Use the **disable** command to disable the Frame Relay CIR monitoring and congestion monitoring features.

The **disable** command dynamically changes the router configuration. These changes will be lost when the router is restarted.

Syntax: disable cir-monitor
| cllm
congestion-monitor
| notify-fecn-source
| throttle-transmit-on-fecn

Example: **disable cir-monitor**

Enable

Use the **enable** command to enable the Frame Relay CIR monitoring and congestion monitoring features.

The **enable** command dynamically changes the router configuration. These changes will be lost when the router is restarted.

Syntax: enable cir-monitor
| cllm
congestion-monitor
| notify-fecn-source
| throttle-transmit-on-fecn

Example: **enable cir-monitor**

List

Use the **list** command to display statistics specific to the data-link layer and the Frame Relay interface.

Syntax: list
- all
- circuit . . .
- lmi
- permanent-virtual-circuits
- pvc-groups

all Displays circuit, management, and PVC statistics on the Frame Relay interface. The output displayed for this command is a combination of the **list lmi** and **list permanent-virtual-circuit** commands.

Example: `list all`

circuit *pvc#*
Displays detailed PVC configuration and statistical information for the specified PVC (pvc#).

Example: `list circuit 347`

```
Circuit name = Valencia

Circuit state        =    Active  Circuit is orphan   =     No
Frames transmitted   =         0  Bytes transmitted   =      0
Frames received      =         0  Bytes received      =      0
Total FECNs          =         0  Total BECNs         =      0
Times congested      =         0  Times Inactive      =      0
CIR in bits/second   =     64000  Potential Info Rate =  56000
Committed Burst (BC) =      1200  Excess Burst (Be)   =  54800
Minimum Info Rate    =     16000  Maximum Info Rate   =  64000
Required             =       Yes  PVC group name      = group1

Compression capable  =       Yes  Operational         =    Yes
R-Rs received        =         0  R-Rs transmitted    =      0
R-As received        =         0  R-As transmitted    =      0
R-R mode discards    =         0  Enlarged frames     =      0
Decompress discards  =         0  Compression errors  =      0
Rcv error discards   =         0

Compression ratio    = 1.72 to 1  Decompression ratio = 1.10 to 1

Current number of xmit frames queued                 =      0
Xmit frames dropped due to queue overflow            =      0
```

Circuit state
Indicates the state of the circuit: inactive, active, or congested. Inactive indicates that the circuit is not available for traffic because either the Frame Relay interface is down or the Frame Relay management entity has not notified the Frame Relay protocol that the circuit is active. Active indicates that data is being transferred. Congested indicates that data flow is being controlled.

Circuit is orphan
Indicates if the circuit is a non-configured circuit learned through LMI management.

Frames/Bytes transmitted
Indicates how many frames and bytes this PVC has transmitted.

Frames/Bytes received
Indicates how many frames and bytes this PVC has received.

Total FECNS
Indicates the number of times that this PVC has been notified of inbound or downstream congestion.

Total BECNS
Indicates the number of times that this PVC has been notified of outbound or upstream congestion.

Times congested
Indicates the number of times that this PVC has become congested.

Times inactive
Indicates the number of times that this PVC was inoperable.

CIR in bits/sec
Indicates the information rate of the PVC between the range 300 bps to 2048000 bps. A value of 0 is also supported.

Potential Info Rate
Indicates the current maximum rate in bits per second at which data will be transmitted for the circuit. The actual data rate will depend on the queue depths and priorities associated with the circuit.

If this field has a value of “Line Speed,” then the maximum data rate is the actual line speed even if the line speed was not configured or was configured incorrectly for this interface.

Committed Burst (BC)
Maximum amount of data, in bits, that the network commits to deliver during a calculated *time interval* (Tc). (Tc=Bc/CIR.)

Excess Burst (Be)
Maximum amount of uncommitted data the router can transmit on a PVC in excess of the Bc during the time interval (Tc).

Minimum Info Rate
Minimum Information Rate. The minimum data rate for a PVC that the router throttles down to when it is notified of congestion.

Maximum Info Rate
Maximum Information Rate. The maximum data rate at which the router transmits for a PVC.

Required
Yes or No. If yes, the PVC is a Required PVC.

PVC group name
If the PVC is a member of a required PVC group, the name appears here; otherwise, “Unassigned” appears.

Compression capable
Indicates whether the circuit can compress data packets.

Operational
Indicates whether compression is active on the circuit. When this is yes, data is being compressed on this link.

R-Rs received
Indicates the number of Reset-Request packets sent by the peer decompressor. A peer decompressor sends a Reset-Request whenever the peer detects that it is out of synch with its peer compressor. If this

number increases rapidly, packets are being lost or corrupted on this circuit.

R-Rs transmitted
Indicates the number of Reset-Request packets sent since compression started on the circuit. If this number increases rapidly, packets are being lost or corrupted on this circuit.

R-As received
Indicates the number of Reset-Acknowledgements received in response to Reset-Requests. The compressor also sends out this packet to signal that it has reset its compression history.

R-As transmitted
This is the number of Reset-Acknowledgements sent to the peer.

R-R mode discards
Indicates the number of compressed data frames that were discarded while waiting for an R-A after sending out an R-R.

Enlarged frames
This is a count of the frames that could not be compressed. Usually an incompressible frame is sent in its uncompressed format within a special compression frame type allowing the compressor and decompressor to remain in synch.

Decompress discards
Indicates the number of compressed frames that were discarded because of decompression errors.

Compression errors
Indicates the number of frames that had compression errors which were transmitted in an uncompressed form.

Rcv error discards
Indicates the number of compressed frames that were discarded because of reception problems.

Compression ratio
Indicates the approximate effectiveness of the compressor.

Decompression ratio
Indicates the approximate effectiveness of the decompressor.

Current number of xmit frames queued
Indicates the number of frames currently queued for this circuit by FR. These frames are waiting for space to become available on the serial device handler transmit queue for this interface.

Xmit frames dropped due to queue overflow
Indicates the number of frames that could not be transmitted for this PVC due to output queue overflow.

lmi Displays statistics relevant to the logical management on the Frame Relay interface.

Example: `list lmi`

```
Management Status:
------------------
  LMI enabled   =   Yes  LMI DLCI         = 1023
  LMI type      =  REV1  LMI Orphans OK   =  Yes
  CLLM enabled  =   Yes  Timer Ty seconds =   11
  Last CLLM cause code = Network congestion - short term (0x02)
  Protocol broadcast   =     Yes  Congestion monitoring     =   Yes
  Emulate multicast    =     Yes  CIR monitoring            =    No
  Notify FECN source   =      No  Throttle transmit on FECN =    No
  PVCs P1 allowed      =      64  Interface down if no PVCs =    No
  Line speed (bps)     =   64000  Interface MTU in bytes    =  2048
  Timer T1 seconds     =      10  Counter N1 increments     =     6
  LMI N2 threshold     =       3  LMI N3 threshold window   =     4
  MIR % of CIr         =      25  IR % Increment            =    12
  IR % Decrement       =      25  DECnet length field       =    No
  Default CIR          =   65636  Default burst size        = 64000
  Default Excess Burst =       0

  Current receive sequence  =  0
  Current transmit sequence =  0
  Total status enquires     =  0    Total status responses    =    0
  Total sequence requests   =  0    Total responses           =    0

  Data compression enabled  =  Yes  Orphan compression              =   No
  Compression PVC limit     = None  Active compression PVCs   =    1

PVC Status:
-----------
  Total allowed  =      64  Total configured  =       3
  Total active   =       0  Total congested   =       0
  Total left net =       0  Total join net    =       0
```

Management Status:

LMI enabled
Indicates if Frame Relay management is active (yes or no).

LMI DLCI
Indicates the management circuit number. This number is either 0 (ANSI default or ITU-T/CCITT) or 1023 (interim LMI REV1).

LMI type
Indicates the type of frame relay management being used, ANSI, ITU-T/CCITT, or LMI Revision 1.

LMI orphans OK
Indicates if all non-configured circuits learned from Frame Relay management are available for use (yes or no).

CLLM enabled
Specifies whether this circuit will throttle transmission on receiving CLLM frames.

Timer Ty seconds
Indicates the value of the CLLM Ty timer. This field is only displayed if CLLM is enabled.

Last CLLM cause code
Indicates the congestion cause code given in the last CLLM message received or ***None*** if no CLLM messages have been received. This field is only displayed if CLLM is enabled.

Protocol broadcast
Indicates if protocols such as IP RIP are able to operate over the Frame Relay interface.

Congestion monitoring
Indicates whether the congestion monitor feature that responds to network congestion is enabled (yes or no).

Emulate multicast
Indicates whether the multicast emulation feature is enabled on each active PVC (yes or no).

CIR monitoring
Indicates whether the circuit monitoring feature that enforces the transmission rate is enabled (yes or no).

PVCs P1 allowed
Indicates the number of allowable PVCs for use with this interface. This number is the maximum number of active, congested, inactive, and removed PVCs that can be supported on the interface.

Interface down if no PVCs
Indicates whether the router considers the interface unavailable when there are no active PVCs.

Line speed (bps)
Indicates the configured data rate of the Frame Relay interface.

Timer T1 seconds
Indicates the frequency with which the Frame Relay interface performs a sequence number exchange with the Frame Relay switch LMI entity.

Counter N1 increments
Indicates the number of T1 timer intervals which must expire before a complete PVC LMI status enquiry is made.

LMI N2 error threshold
Indicates the number of management event errors occurring within the N3 window that will cause a reset of the Frame Relay interface.

LMI N3 error threshold window
Indicates the number of monitored management events used to measure the N2 error threshold.

MIR % of CIR
Minimum IR, expressed as a percentage of CIR.

IR % Increment
Percentage by which the router increments the IR each time it receives a frame without BECN until it reaches the maximum IR.

IR % Decrement
Percentage by which the router decrements the IR each time it receives a frame that contains BECN until it reaches the minimum IR.

DECnet length field
Indicates whether or not the DECnet length field feature is enabled. Some Frame Relay DECnet Phase IV implementations require a length field between the Frame Relay multiprotocol encapsulation header and the DECnet packet. A length field is inserted if the DECnet length field feature is enabled.

Default CIR
Specifies the default CIR for this interface.

Default Burst Size
Specifies the default burst size for this interface.

Default Excess CIR
Specifies the default excess burst size for this interface.

Current receive sequence
Indicates the current receive sequence number that the Frame Relay interface has received from the Frame Relay management entity.

Current transmit sequence
Indicates the current transmit sequence number that the Frame Relay interface has sent to the Frame Relay management entity.

Total status enquiries
Indicates the total number of status enquiries that the Frame Relay interface has made of the Frame Relay management entity.

Total status responses
Indicates the total number of responses that the Frame Relay interface has received from the Frame Relay management entity in response to status enquiries.

Total sequence requests
Indicates the total number of sequence number requests that the Frame Relay interface has sent to the Frame Relay management entity.

Total responses
Indicates the total number of sequence number responses that the Frame Relay interface has received from the Frame Relay management entity.

Data compression enabled
Indicates whether data compression is enabled on this interface.

Orphan compression
Indicates whether orphan circuits on this interface will have data compression enabled.

Note: Enabling compression on orphan circuits will decrease the number of available compression contexts available for the native PVCs on the device.

Compression PVC limit
Specifies the maximum number of PVCs that can compress data on this interface.

Active compression PVCs
Specifies the number of PVCs currently compressing data on this interface.

PVC Status:

Total allowed
Indicates the number of allowable PVCs (including orphans) whose state is active, congested, removed, or inactive for use with this interface.

Total configured
Indicates the total number of currently configured PVCs for this interface.

Total active
Indicates the number of active PVCs on this interface.

Total congested
Indicates the number of PVCs that are throttled down because of congestion within the network.

Total left net
Indicates the total number of PVCs that have been removed from the network.

Total join net
Indicates the total number of PVCs that have been added to the network.

permanent-virtual-circuit
Displays general link-layer statistics and configuration information for all configured PVCs on the Frame Relay interface.

Example: **`list permanent-virtual-circuit`**

```
                          Orphan   Type/    Frames       Frames
 Circuit#     Circuit Name  Circuit  State  Transmitted   Received
 --------  ----------------  -------  -----  -----------   ---------
     16    Valencia             No    @*P/A            2         1
     17    Raleigh              No    @#P/A           15        14
     18    Boston               No    &#P/A            0         0
     19    Orlando              No     *P/A            0         0
     20    Port Royal           No      P/A            0         0
     21    New York             No     @P/A            2         0

    A - Active     I - Inactive   R - Removed   P - Permanent  C - Congested
    * - Required                  # - Required and belongs to a PVC group
    @ - Data compression capable but not operational
    & - Data compression capable and operational
```

Circuit#	Indicates the number of the PVC.
Circuit Name	Name of the circuit, an ASCII string.
Orphan Circuit	Indicates whether the PVC is a non-configured circuit (yes or no).
Type/State	Indicates the state of the circuit, A (active), I (inactive), P (permanent), C (congested), or R (removed).
Frames Transmitted	Indicates how many frames this PVC has transmitted.
Frames Received	Indicates how many frames this PVC has received.

pvc-groups
Displays required PVC group information for all required PVC groups. For each group this consists of the group name, the circuits in the group and the state (active, inactive, or removed) of each circuit.

Example: **`list pvc-groups`**

```
Group name              Circuits in group    Circuit status
----------              -----------------    --------------

group1                            16             active
                                  44             inactive
                                 240             removed
```

LLC

Use the **LLC** command to access the LLC monitoring prompt. LLC commands are entered at this new prompt. See “LLC Monitoring Commands” on page 25-1 for an explanation of each of these commands.

Syntax: llc

Example: **llc**

```
Circuit number to monitor [0]?
LLC user monitoring
LLC>
```

Note: The LLC command is supported only if APPN is in the software load.

Set

Use the **set** command to set the values for Committed Information Rate (CIR), Committed Burst Rate, and Excess Burst Rate for the specified PVC. You also can set values for IR adjustment rates.

Changes made with this command do not affect the configuration data, they are in effect only until the router is restarted.

Syntax: set circuit . . .
ir-adjustment . . .

circuit

Sets the values for Committed Information Rate (CIR), Committed Burst Rate, and Excess Burst Rate for the specified PVC.

ir-adjustment *increment-% decrement-% minimum-IR*

Sets the minimum information rate (IR) and the percentages for incrementing and decrementing the IR in response to network congestion.

The minimum IR, expressed as a percentage of CIR, is the lower limit of the information rate. The minimum percentage is 1 and the maximum percentage is 100. The default is 25.

When network congestion clears, the information rate is gradually incremented by the IR adjustment increment percentage until the maximum information rate is reached. The minimum percentage is 1 and the maximum percentage is 100. The default is 12.

When network congestion occurs, the information rate is decremented by the IR adjustment decrement percentage each time a frame containing BECN is received until the minimum information rate is reached. The minimum percentage is 1, and the maximum percentage is 100. The default is 25.

Example: **set ir-adjustment**

```
IR adjustment % increment [12]?
IR adjustment % decrement [25]?
Minimum IR as % of CIR [25]?
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **exit**

Frame Relay Interfaces and the GWCON Interface Command

While Frame Relay interfaces have a console process for monitoring purposes, the router also displays complete statistics for installed interfaces when you use the **interface** command from the GWCON environment. (For more information on the **interface** command, refer to Chapter 6, “The GWCON (Monitoring) Process and Commands” on page 6-1)

Statistics Displayed For Frame Relay Interfaces

Statistics similar to the following are displayed when you execute the **interface** command from the GWCON environment for Frame Relay interfaces:

```
+interface 1
                                Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
1  1   FR/0         81620   5D          1          0            0

  Frame Relay MAC/data-link on SCC Serial Line interface

  Adapter cable:               V.35 DTE  RISC Microcode Revision:
    1

   V.24 circuit: 105 106 107 108 109 125 141
   Nicknames:    RTS CTS DSR DTR DCD RI  LL
   PUB 41450:    CA  CB  CC  CD  CF  CE
   State:        ON  ON  ON  ON  ON  OFF OFF

  Line speed:              unknown
  Last port reset:         5 hours, 8 minutes, 11 seconds ago

  Input frame errors:
   CRC error                          0   alignment (byte length)
   missed frame                       0   too long (>  2062 bytes) 0
   aborted frame                      0   DMA/FIFO overrun         0
   L & F bits not set                 0
  Output frame counters:
   DMA/FIFO underrun errors           0   Output aborts sent       0
```

Nt Indicates the interface number as assigned by software during initial configuration.

Nt' Indicates the interface number as assigned by software during initial configuration.

Note: For FR dial circuit interfaces, Nt' is different from Nt. Nt' indicates the base interface (ISDN) that the dial circuit is running over.

Interface
Indicates the type of interface and its instance number. Frame relay has a FR designation.

CSR Indicates the memory location of the control status register for the Frame Relay interface.

Vec Indicates the vector number for the Frame Relay interface.

Self-test Passed
Indicates the total number of times the Frame Relay interface passed self-test.

Self-test Failed
Indicates the total number of times the Frame Relay interface failed self-test.

Maintenance Failed
: Indicates the total number of times the interface was unable to communicate with Frame Relay management.

V.24 circuit, Nicknames, and State
: The circuits, control signals, pin assignments and their state (ON or OFF). Note: The symbol - - - in console output indicates that the value or state is unknown.

Line speed
: The transmit clock rate.

Last port reset
: The length of time since the last port reset.

Input frame errors:

CRC error
: The number of packets received that contained checksum errors and as a result were discarded.

Alignment
: The number of packets received that were not an even multiple of 8 bits in length and a result were discarded.

Too short
: The number of packets that were less than 2 bytes in length and as a result were discarded.

Too long
: The number of packets that were greater than the configured size, and as a result were discarded.

Aborted frame
: The number of packets received that were aborted by the sender or a line error.

DMA/FIFO overrun
: The number of times the serial interface could not send data fast enough to the system packet buffer memory to receive them from the network.

Missed frame
: When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.

L & F bits not set
: On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the `L & F bits not set` counter is incremented, and the buffer is cleared for reuse.

  Note: It is unlikely that the `L & F bits not set` counter will be affected by traffic.

Output frame counters:

DMA/FIFO underrun errors
: The number of times the serial interface could not retrieve data fast enough from the system packet buffer memory to transmit them to the network.

Output aborts sent
: The number of transmissions that were aborted as requested by upper-level software.

Statistics similar to the following are displayed for Frame Relay dial circuits when you execute the interface command from the GWCON environment:

```
+interface 4

                                              Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec    Passed      Failed       Failed
4  3    FR/0          81640   5C             0             4              0

  Frame Relay MAC/data-link on ISDN Basic Rate interface
```

Chapter 33. Using and Configuring Point-to-Point Protocol Interfaces

This chapter describes how to configure the Point-to-Point Protocol for interfaces on the device. Sections in this chapter include:

See Chapter 35, "Using and Configuring the Multilink PPP Protocol" on page 35-1 and Chapter 36, "Monitoring Multilink Protocol (MP)" on page 36-1 for information about using the Multilink PPP Protocol.

PPP Overview

PPP provides a method for transmitting protocol datagrams at the Data Link Layer over serial point-to-point links. PPP provides the following services:

- Link Control Protocol (LCP) to establish, configure, and test the link connection.
- Encapsulation protocol for encapsulating protocol datagrams over serial point-to-point links.
- Authentication protocols (APs) to validate the identity of a peer (remote) unit, and to submit your own identity to the peer for validation.
- Network Control Protocols (NCPs) for establishing and configuring different network layer protocols. PPP allows the use of multiple network layer protocols.

Figure 33-1 on page 33-2 shows some examples of point-to-point serial links.

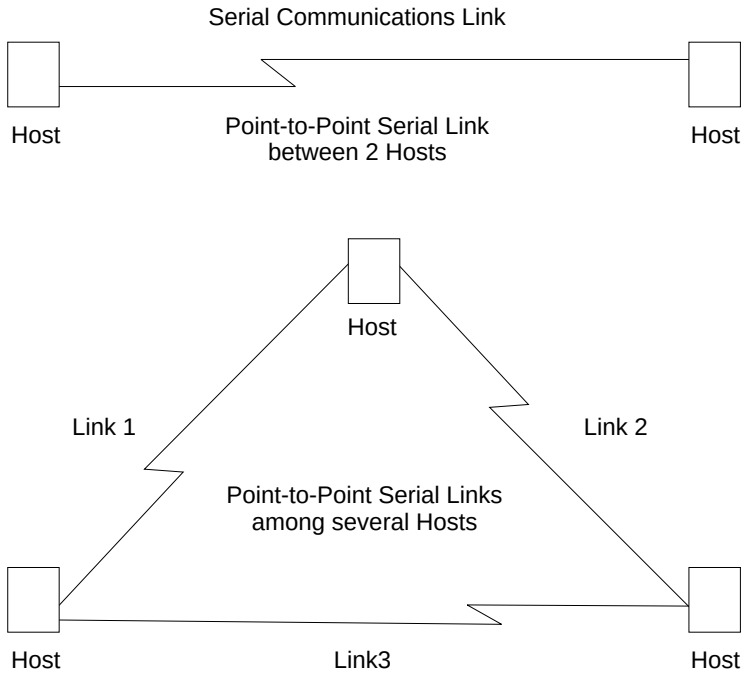


Figure 33-1. Examples of Point-to-Point Links

PPP currently supports AppleTalk Control Protocol (ATCP), DECnet Protocol Control Protocol (DNCP), Banyan VINES Control Protocol (BVCP), bridging protocols (BCP, NBCP, and NBFCP), Internet Protocol Control Protocol (IPCP), IPX Control Protocol (IPXCP), APPN HPR Control Protocol (APPN HPRCP), APPN ISR Control Protocol (APPN ISRCP), and OSI Control Protocol (OSICP).

Each end starts by sending LCP packets to configure and test the data link. After the link has been established, PPP sends NCP packets to choose and configure one or more network layer protocols. After network layer protocols have been configured, datagrams from each network layer can be sent over the link. The next sections explain these concepts in more detail.

PPP Data Link Layer Frame Structure

PPP transmits data frames that have the same structure as High-level Data Link Control (HDLC) frames. PPP uses a byte-oriented transmission method with a single-frame format for all data and control exchanges. Figure 33-2 illustrates the PPP frame structure and is followed by a detailed description of each field.

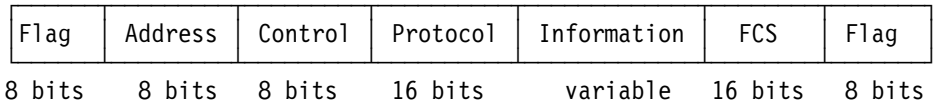


Figure 33-2. PPP Frame Structure

Flag Fields

The flag field begins and ends each frame with a unique pattern of 01111110. Generally a single flag ends one frame and begins the next. The receiver attached to the link continuously search for the flag sequence to synchronize the start of the next frame.

Address Field

The address field is a single octet (8 bits) and contains the binary sequence 11111111 (0xff hexadecimal). This is known as the All-Station Address. PPP does not assign individual station addresses.

Control Field

The control field is a single octet and contains the binary sequence 00000011 (0x03 hexadecimal). This sequence identifies the Unnumbered Information (UI) command with the P/F bit set to zero.

Protocol Field

The protocol field is defined by PPP. The field is 2 octets (16 bits) and its value identifies the protocol datagram encapsulated in the Information field of the frame.

Protocol field values in the range '0xC000'–'0xFFFF' indicate Layer 3 data (protocol datagrams) such as LCP, PAP, CHAP, SPAP, and CCP. Values in the range '8000'–'BFFF' indicate that the datagrams belong to the Network Control Protocols (NCP). Values in the range '0'–'3FFF' identify the network protocol of specific datagrams.

Information Field

The information field contains the datagram for the protocol specified in the protocol field. This is zero or more octets.

When the protocol type is LCP, exactly one LCP packet is encapsulated in the information field of PPP Data Link Layer frames.

Frame Check Sequence (FCS) Field

The frame check sequence field is a 16-bit cyclic redundancy check (CRC).

PPP links can negotiate the use of various options which may modify the basic frame format; the description below applies to the frame format prior to any such modifications. PPP LCP packets are always sent in this format as well, regardless of negotiated options, so that LCP packets can be recognized even when there is a loss of synchronization on the line.

The router supports two such options: Address and Control Field Compression (ACFC) and Protocol Field Compression (PFC). These are described in detail in a later section.

The PPP Link Control Protocol (LCP)

PPP's Link Control Protocol (LCP) establishes, configures, maintains, and terminates the point-to-point link. This process is carried out in four phases:

1. Before exchanging any network layer datagrams, PPP first opens the connection through an exchange of LCP configuration packets. As part of this negotiation process, the PPP processes at each end of the link agree on various basic link level parameters such as the maximum packet size that can be transferred and whether the ends must use an authentication mechanism to identify themselves to their peers before carrying network traffic.

   If this negotiation is unsuccessful, the link is considered to be "down" and incapable of carrying any network traffic. If the negotiation is successful, LCP goes to an "Open" state and PPP goes on to the next phase.

2. After LCP successfully reaches an Open state, the next step in establishing the link is to perform authentication where each end of the link identifies itself to the other end using the “authentication protocol” that the other end dictated as part of the LCP negotiation.

   If authentication fails, the link is marked “down” and cannot carry any network traffic. If authentication succeeds or if authentication is not required, the PPP link moves to the next phase.

3. After authentication is negotiated, the peers negotiate encryption for the link. After authentication phase is complete, the router negotiates the use of encryption using Encryption Control Protocol (ECP) packets where each end of the link negotiates which encryption algorithm will be used to encrypt the data over this PPP link. If ECP did not reach “Open” state then the link is marked “down” and cannot carry any network traffic. If ECP successfully reaches “Open” state, or if encryption is not required, the PPP link moves to the next phase, NCP negotiation (except ECP, which is technically also an NCP). The link is considered to be “open” or “up” at this time, though it cannot yet carry layer-3 protocol datagrams.

4. Once the link is open, the router negotiates the use of various layer-3 protocols (for example, IP, IPX, DECnet, Banyan Vines) using Network Control Protocol (NCP) packets. Each layer-3 protocol has its own associated network control protocol. For example IP has IPCP and IPX has IPXCP. The basic format and mechanisms for all these NCP packets is the same for all protocols, and is basically a superset of the LCP mechanisms as described later in this section.

   Each layer-3 protocol is negotiated independently. When a particular NCP successfully negotiates, the link is “up” for that protocol's traffic. As with LCP, configuration information can be exchanged as part of this negotiation; for example, IPCP can exchange IP addresses or negotiate the use of "Van Jacobson IP header compression".

   As with LCP, it is possible for an NCP to fail to negotiate successfully with its peer. This might happen because the peer does not support a particular protocol or because some configuration option was unacceptable. If an NCP fails to reach the “Open” state, no layer-3 protocol packets can be exchanged for that protocol even though other layer-3 protocols are successfully passing traffic across the PPP link.

5. Finally, LCP has the ability to terminate the link at any time. This is usually done at the request of the user but may occur for other reasons such as: an administrative closing of the link, idle timer expiration, or failure to reauthenticate on a CHAP rechallenge.

For complete details about PPP LCP, authentication, and the general NCP negotiation mechanisms, consult RFCs 1331, 1334, 1570, and 1661.

LCP Packets

LCP packets are used to establish and manage a PPP link and can be loosely divided into three categories:

- *Link establishment packets* that exchange configuration information and establish the link.

- *Link termination packets* that shut down the link or signal that a link is not accepting connections at a particular time. They also can be used to signal that a particular protocol is unrecognized (for example, during NCP negotiations).
- *Link maintenance packets* that monitor and debug a link.

Exactly one LCP packet is encapsulated in the information field of PPP Data Link Layer frames. In the case of LCP packets, the protocol field reads “Link Control Protocol” (C021 hexadecimal). Figure 33-3 illustrates the structure of the LCP packet and is followed by a detailed description of each field.

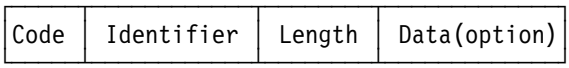


Figure 33-3. LCP Frame Structure (in PPP Information Field)

Code

The code field is one octet in length and identifies the type of LCP packet. The codes in Table 33-1 distinguish the packet types. They are described in more detail in later sections.

Table 33-1. LCP Packet Codes

Code	Packet Type
1	Configure-Request (Link Establishment)
2	Configure-Ack (Link Establishment)
3	Configure-Nak (Link Establishment)
4	Configure-Reject (Link Establishment)
5	Terminate-Request (Link Termination)
6	Terminate-Ack (Link Termination)
7	Code-Reject (Link Establishment)
8	Protocol-Reject (Link Establishment)
9	Echo-Request (Link Maintenance)
10	Echo-Reply (Link Maintenance)
11	Discard-Request (Link Maintenance)

Identifier

The identifier field is one octet in length and is used to match packet requests to replies.

Length

The length field is two octets in length and indicates the total length (that is, including all fields) of the LCP packet.

Data (Option)

The data field is zero or more octets as indicated by the length field. The format of this field is determined by the code.

NCP packets are structured identically to LCP packets and are distinguished by having different PPP “Protocol” values. Each LCP packet type (distinguished by the code field) has the same meaning for each NCP, though an individual NCP may not implement all possible LCP packet types. NCPs normally implement all of the link establishment type packets that LCP defines. They may implement some of the

additional LCP packet types, and they also may define additional packet types beyond what LCP uses. Unlike LCP packets, the structure of an NCP frame may be modified according to options negotiated by LCP during the link establishment phase.

Link Establishment Packets

Link Establishment Packets establish and configure a point-to-point link including the following packet types:

Configure-Request

LCP packet code field is set to 1. LCP transmits this packet type when it wants to open a point-to-point link. Upon receiving a Configure-Request, a peer station's LCP entity sends an appropriate reply, depending on whether it is ready to process packets .

Configure-Ack

LCP packet code field is set to 2. The peer transmits this packet type when every configuration option in a Configure-Request packet is acceptable. Upon receiving the Configure-Ack (ack = acknowledgment), the originating station checks the Identifier field. This field must match the one from the last-transmitted Configure-Request or the packet is invalid.

Both ends send Configure-Request and both ends must receive a Configure-Ack before the link opens. Options negotiated for one direction may differ from that negotiated for the other direction. There is no "master-slave" relationship. Rather, each end works symmetrically.

Configure-Nak

LCP packet code field is set to 3. The peer transmits this packet type when some part of the configuration option in a Configure-Request packet is unacceptable. The Identifier field is copied from the received Configure-Request and the Data (option) field is filled with the received unacceptable configuration options. The Identifier field must match the one from the last-transmitted Configure-Request or the packet is invalid and is discarded.

When the originator receives a Configure-Nak packet, a new Configure-Request packet is sent that includes modified, acceptable configuration options.

Configure-Reject

LCP packet code field is set to 4. The peer transmits this packet type when some part of the configuration options in a Configure-Request packet is unacceptable. The Identifier field is copied from the received Configure-Request and the Data (option) field is filled with the received unacceptable configuration options. The Identifier field must match the one from the last-transmitted Configure-Request or the packet is invalid and is discarded.

When the originator receives a Configure-Reject packet, a new Configure-Request packet is sent that does not include any of the configuration options received in the Configure-Reject packet.

Code-Reject

LCP packet code field is set to 7. The transmission of this packet type indicates that the LCP "code" field on a received packet is not recognized as a valid value. While this can indicate an error, it also can indicate that the peer does not implement some feature that you are trying to use.

Protocol-Reject

LCP packet code field is set to 8. The transmission of this packet type indicates that a PPP frame has been received that contains an unsupported or unknown protocol (the PPP “protocol” field was unrecognized for some packet). This usually occurs if you try to negotiate some NCP for a protocol that the other end doesn't support. For example, if DECNet CP (DNCP) sends a Config-Request and the other end does not know about DECNet, the other end replies with an LCP Protocol-Reject on DNCP. Upon receiving a Protocol-Reject packet, the link stops transmitting the incorrect protocol.

Note: NCP packet types and structure are the same as LCP, although there are a few additional “code” fields associated with some NCPs.

Link Termination Packets

Link Termination Packets terminate a link and include the following packet types:

Terminate-Request

LCP packet code field is set to 5. LCP transmits this packet type when a point-to-point link needs to be closed. These packets are sent until a Terminate-Ack packet is sent back, or until a retry counter is exceeded while waiting for an Ack.

Terminate-Ack

LCP packet code field is set to 6. Upon receiving a Terminate-Request packet, this packet type must be transmitted with the code field set to 6. Reception of an Terminate-Ack packet that was not expected indicates that the link has been closed.

Link Maintenance Packets

Link Maintenance Packets manage and debug a link, and include the following packet types:

Echo-Request and Echo-Reply

LCP packet code fields are set to 9 and 10 respectively. LCP transmits these packet types in order to provide a Data Link Layer loopback mechanism for both directions on the link. This feature is useful, for example, in debugging a faulty link to determine link quality. These packets are sent only when the link is in the Open state.

Discard-Request

LCP packet code field is set to 11. LCP transmits this packet type to provide a data sink for Data link Layer testing. A peer that receives a Discard-Request ***must*** throw away the packet. This is useful in debugging a link. These packets are sent only when the link is in the Open state.

PPP Authentication Protocols

PPP authentication protocols provide a form of security between two nodes connected via a PPP link. If authentication is required on a box, then immediately after the two boxes successfully negotiate the use of the link at the LCP layer (LCP packets are exchanged until LCP goes into an “open” state), they go into an “authentication” phase where they exchange authentication packets. A box is neither able to carry network data packets nor negotiate the use of a network protocol (NCP traffic) until authentication negotiation completes successfully.

There are different authentication protocols in use: PAP (Password Authentication Protocol) and CHAP (Challenge/Handshake Authentication Protocol). These are described in detail in RFC 1334, and briefly described later in this section. On remote dial-in access ports, a third authentication protocol is available. This is SPAP (Shiva Password Authentication Protocol), which is a Shiva proprietary protocol. See “Shiva Password Authentication Protocol (SPAP)” on page 33-9 for more information.

Whether a box requires the other end to authenticate itself (and if so, with what protocol) is determined during the LCP negotiation phase. Authentication could be considered to “fail” even at the link establishment phase (LCP negotiation), if one end does not know how, or refuses to use, the authentication protocol the other end requires.

Each end of a link sets its own requirements for how it wants the other end to authenticate itself. For example, given two routers “A” and “B,” connected over a PPP link, side A may require that B authenticate itself to A using PAP, and side B may require that A similarly identify itself using CHAP. It is valid for one end to require authentication while the other end requires none.

In addition to initial authentication during link establishment, with some protocols an authenticator may demand that the peer reestablish its credentials periodically. With CHAP, for example, a rechallenge may be issued at any time by the authenticator and the peer must successfully reply - or lose the link.

If more than one authentication protocol is enabled on a link, the router initially attempts to use them in the priority order that you specify :

1. CHAP
2. PAP
3. SPAP

Note: SPAP is only available on interfaces that have IBM DIALs Dial-In circuits configured.

If the remote side responds to the authentication request with NAK and suggests an alternative, the router uses the alternative provided it is enabled on the link. If the remote side continues responding to the router's suggestions with a NAK but does not provide an alternative that the router has enabled, the link is terminated.

Password Authentication Protocol (PAP)

The Password Authentication Protocol (PAP) provides a simple method for the peer to establish its identity using a two-way handshake. This is done only upon initial link establishment. Following link establishment, the peer sends an ID/Password pair to the authenticator until authentication is acknowledged or the connection is terminated. Passwords are sent over the circuit “in the clear,” and there is no protection from playback or repeated trial and error attacks. The peer controls the frequency and timing of the attempts.

Challenge-Handshake Authentication Protocol (CHAP)

The Challenge-Handshake Authentication Protocol (CHAP) is used to periodically verify the identity of the peer using a three-way handshake. This is done upon initial link establishment, and *may* be repeated anytime after the link has been established. After the initial link establishment, the authenticator sends a "challenge" message to the peer. The peer responds with a value calculated using a "one-way hash" function. The authenticator checks the response against its own calculation of the expected hash value. If the values match, the authentication is acknowledged; otherwise the connection is terminated.

Shiva Password Authentication Protocol (SPAP)

Note: SPAP is only available on interfaces that have IBM DIALs Dial-In circuits configured.

The Shiva Password Authentication Protocol (SPAP) provides a simple method for the peer to establish its identity using a 2-way handshake similar to PAP. After the Link Establishment phase is complete, an Id/Password is repeatedly sent by the peer to the authenticator until authentication is acknowledged, the connection is terminated, or a retry counter expires.

SPAP is a moderately strong authentication protocol that uses a proprietary encryption algorithm for the password. It offers additional function in concert with authentication:

- The ability to change a password.
- The ability for the router to send a configurable banner requiring acknowledgment from the client after password authentication.
- The ability to use callback as an additional security feature.

Configuring PPP Authentication

The following sections describe configuring PPP authentications for two situations:

- Configuring the 2210 to authenticate a remote device.
- Configuring the 2210 to be authenticated by a remote device.

These two situations are independent. You can do one or the other.

Configuring a PPP Interface to Authenticate a Remote Device

To authenticate a remote device or dial-in client:

1. Enable authentication on the PPP interface
   - At the `Config>` prompt, enter the **network** command to select the PPP interface to configure.
   - At the `PPP Config>` prompt, enable the authentication protocol you want to use.

     You can use any of the following protocols:

     – PAP
     – CHAP
     – SPAP

     Note: SPAP is only available on interfaces that have IBM DIALs Dial-In circuits configured.

2. Decide whether to authenticate locally or through an authentication server.
   - To authenticate locally, enter the name and password into the PPP user database.

     At the `Config>` prompt, use the **add ppp_user** command. See “Add” on page 3-12 for more information.

     A 2210 maintains a single PPP user database. When the remote router or device sends its name and password to the device during the authentication phase, the device checks to see if that name and password are in the PPP user database.
   - To authenticate through an authentication server using TACACS, TACACS+, or RADIUS, you must configure the device to reach the authentication server and the name and password must be in the server's database. Refer to Chapter 20, “Configuring Local or Remote Authentication” on page 20-1.

Configuring a PPP Interface to be Authenticated by a Remote Device

To configure the device to be authenticated by a remote device or dial-in client, configure the device's name and password:

1. At the `Config>` prompt, select the interface you are configuring using the **network** command.
2. At the PPP `Config>` prompt, type the **set name** command and provide the name and password that the device will use to identify itself to the remote router or device during the authentication phase.

Attention: Do not use the following commands unless you want the device to perform authentication as described in Chapter 20, “Configuring Local or Remote Authentication” on page 20-1.

- **enable pap**
- **enable chap**
- **enable spap**

  Note: SPAP is only available on interfaces that have IBM DIALs Dial-In circuits configured.

Configuring PPP Callback

Callback is a PPP feature associated with single user dial-in solutions. It attempts to accomplish two objectives. These objectives are:

- Callback can be used as a form of security. When used in this way, callback is generally referred to as required callback. When required callback is negotiated the user will be dialed back at a predetermined number. Only then will the PPP link be allowed to come up.
- Callback can also be implemented as a toll-saver feature. When used in this way, callback is generally referred to as roaming callback. Unlike required callback, roaming callback is requested by the client. The primary function of roaming callback is to bill the company maintaining the DIALs Server the toll charges instead of the user.

Callback is supported only on dial-in dial circuits over V.34 or ISDN networks.

Example 1: Required callback enabled

```
Config>add PPP
Enter user name: []? sallydoe
Password:
Enter password again:
Is this a Single-User or a Network? (Single-User, Network): [Single-User]

IP address for user sallydoe [0.0.0.0]?
Enter HostName: []?
Give 'sallydoe' default time allotted ? (Yes, No): [Yes]
Enable Callback for 'sallydoe' ? (Yes, No): [No] yes
Type of Callback (Roaming Callback, Required Callback): [Roaming Callback] Requ
Dialback number for this user []? 555-1234
Will 'sallydoe' be able to dial-out ? (Yes, No): [No]
Enable encryption for this user/port (y/n) [No]:

PPP User Name: sallydoe
Type: Single User
User IP Address: Interface Default
SubNetMask: 255.255.255.255
Hostname: <undefined>
Time-Alotted: Box Default
Call-Back Type: Required Callback
Phone Number: 543-3186
Dial-Out: Not Enabled
Encryption: Not Enabled

Is information correct? (Yes, No, Quit): [No] yes
```

Example 2: Callback disabled

```
Config>add PPP
Enter user name: []? sallydoe
Password:
Enter password again:
Is this a Single-User or a Network? (Single-User, Network): [Single-User]

IP address for user sallydoe [0.0.0.0]?
Enter HostName: []?
Give 'no callback' default time allotted ? (Yes, No): [Yes]
Enable Callback for 'no callback' ? (Yes, No): [No]
Will 'no callback' be able to dial-out ? (Yes, No): [No]
Enable encryption for this user/port (y/n) [No]:


PPP User Name: no callback
Type: Single User
User IP Address: Interface Default
SubNetMask: 255.255.255.255
Hostname: <undefined>
Time-Alotted: Box Default
Call-Back Type: Not Enabled
Dial-Out: Not Enabled
Encryption: Not Enabled

Is information correct? (Yes, No, Quit): [No] yes
```

Example 3: Roaming callback enabled

```
Config>add PPP roaming_callback
Password:
Enter password again:
Is this a Single-User or a Network? (Single-User, Network): [Single-User]

IP address for user roaming_callback [0.0.0.0]?
Enter HostName: []?
Give 'roaming_callback' default time allotted ? (Yes, No): [Yes]
Enable Callback for 'roaming_callback' ? (Yes, No): [No] yes
Type of Callback (Roaming Callback, Required Callback): [Roaming Callback]

Will 'roaming_callback' be able to dial-out ? (Yes, No): [No]n
Enable encryption for this user/port (y/n) [No]:


PPP User Name: roaming_callback
Type: Single User
User IP Address: Interface Default
SubNetMask: 255.255.255.255
Hostname: <undefined>
Time-Alotted: Box Default
Call-Back Type: Roaming Callback
Dial-Out: Not Enabled
Encryption: Not Enabled

Is information correct? (Yes, No, Quit): [No]yes
```

The PPP Network Control Protocols

PPP has a family of Network Control Protocols (NCPs) for establishing and configuring different network layer protocols. The NCPs are responsible for configuring, enabling, and disabling the network layer protocols on both ends of the point-to-point link. NCP packets cannot be exchanged until LCP has opened the connection and the link reaches the OPEN state.

PPP supports the following Network Control Protocols:

- AppleTalk Control Protocol (ATCP)
- Banyan VINES Control Protocol (BVCP)
- Bridging protocols (BCP, NBCP, and NBFCP),
- DECnet Control Protocol (DNCP)
- IP Control Protocol (IPCP)
- IPX Control Protocol (IPXCP)
- OSI Control Protocol (OSICP)
- APPN High Performance Routing Control Protocol (APPN HPRCP)
- APPN Intermediate Session Routing Control Protocol (APPN ISRCP)

AppleTalk Control Protocol

ATCP is specified in Request for Comments (RFC) 1378. IBM's implementation of ATCP supports the AppleTalk-Address option. The implementation supports both full router mode and half router mode. For additional information, refer to "AppleTalk over PPP" in *Protocol Configuration and Monitoring Reference Volume 2 for Nways Multiprotocol Routing Services Version 2.1*

Banyan VINES Control Protocol

RFC 1763 describes BVCP. IBM's implementation of BVCP does not support any options.

Bridging Protocols

Bridging Control Protocol (BCP) is specified in RFC 1220. IBM's implementation of BCP supports the IEEE 802.5 Line Identification Option and the Tinygram Compression Option.

NetBIOS Control Protocol (NBCP) is a proprietary NCP developed by Shiva Corporation and used by the IBM Dial In Access to LAN Client for OS/2, DOS and Windows for single-user dial-in. NBCP is used to transport NetBIOS and LLC/802.2 bridged traffic from these clients, dialed into a 2210 DIALs Server, onto an attached LAN. IBM's implementation of NBCP supports the MAC-Address and NetBIOS Name Projection options.

NetBIOS Frame Control Protocol (NBFCP) is specified in RFC 2097. NBFCP is used by Microsoft Windows 95 and Windows NT Dial-Up Networking clients for single-user dial-in. NBFCP is used to transport NetBIOS bridged traffic from these clients, dialed into a 2210 DIALs Server, onto an attached LAN. IBM's implementation of NBFCP supports the Name-Projection, Peer-Information and IEEE-MAC-Address-Required options.

DECnet Control Protocol

DNCP is specified in RFC 1376. IBM's implementation does not support any DNCP options.

IP Control Protocol

IPCP is specified in RFC 1332. IBM's implementation supports the following options:

- Van Jacobsen IP Header Compression as described in RFC 1144.
- IP Address

  The router can send its IP address, as well as accept an IP address, from a peer, or supply an IP address to a peer, if requested. If the router is configured to "Send Our Address" on a particular interface, and that interface has a valid, numbered IP address, then IPCP sends the address in its initial Configure-Request as option 3 (IP Address). IPCP also sends its address if the peer sends a Configure NAK with 0.0.0.0 for option 3 (IP Address), if a valid numbered address is configured for that PPP interface. IPCP will not send an unnumbered address to its peer.

  A peer may specify its address (referred to as "Client Specified"), or request an address from the router by sending 0.0.0.0 for Option 3 in its initial Configure Request. The router may obtain this address from: the authenticated user profile (referred to as "User ID"), the interface itself (referred to as "Interface"), or the Dynamic Host Configuration Protocol (referred to as "Proxy DHCP"). Any one of these four methods for specifying the peer's IP address may be disabled or enabled at the 2210 level. For more information on enabling and disabling these items, see Chapter 37, "Using and Configuring a Dial-In Access to LANs (DIALs) Server" on page 37-1.

The router automatically adds a static route directed to the PPP interface for the address that is successfully negotiated, allowing data to be routed properly to the dial-in client. When the IPCP connection is ended for any reason, this static route is subsequently removed. By default, the net mask for this route is 255.255.255.255 (hostroute), however if a net mask is specified in the authenticated user's profile (see "Configuring PPP Authentication" on page 33-9) a net mask other than this may be used to allow routing to more than a single host across the PPP link (RIP or other routing protocols could also be used to discover routes if desired).

IPX Control Protocol

IPXCP is specified in RFC 1552. IBM's implementation does not support any IPXCP options.

OSI Control Protocol

OSICP is specified in RFC 1377. IBM's implementation of OSICP does not support any options.

APPN HPR Control Protocol

Advanced Peer-to-Peer Networking (APPN) High Performance Routing (HPR) control protocol is specified in RFC 2043. No options are negotiated for this control protocol.

APPN ISR Control Protocol

Advanced Peer-to-Peer Networking (APPN) Intermediate Session Routing (ISR) control protocol is specified in RFC 2043. No options are negotiated for this control protocol.

Overview of Encryption

Nways devices support encryption as described in RFCs 1968 and 1969. The objective of encryption is to transform data into an unreadable form to ensure privacy. The ***encrypted*** data needs to be decrypted to get the original data. A method of encryption and decryption is called an ***encryption algorithm***. Encryption algorithms use a key to control encryption and decryption.

The Encryption Control Protocol is used in the router to negotiate the use of encryption on the point-to-point links communicating using PPP protocol . The Encryption Control Protocol provides a generalized mechanism to negotiate which encryption and decryption algorithms will be used over a PPP link. Different encryption algorithms can be negotiated in each direction of the PPP link.

Nways devices support Data Encryption Standard (DES) Cipher Block Chaining (CBC) mode. DES is a symmetric encryption standard that uses a 56-bit key for encryption and decryption. Unlike compression, the router encrypts in both directions of the link, because encrypting in only one direction is a security risk. The link will be terminated whenever ECP cannot negotiate encryption algorithms in both directions.

Configuring Encryption

In order to configure the device to use encryption at the data link layer, you should:

1. Set the encryption keys for remote devices and local PPP interfaces.

   Set the encryption key for the remote device using the **add ppp-user** command at the `Config >` prompt (see "Add" on page 3-12).

   Set the encryption key for the local PPP interface using the **set name** command (see "Set" on page 33-23).

2. Configure individual PPP links to use Encryption Control Protocol (ECP) by using the **enable ecp** command at the `PPP Config>` prompt (see "Enable" on page 33-18).

You can also disable encryption, change the encryption key for a user, list the status of encryption, or set the name and encryption key the device uses when requesting encryption. For information about

- Disabling encryption, see the **disable ecp** command in "Disable" on page 33-17.
- Changing the user's encryption key, see the **change ppp-user** command in "Change" on page 3-18.
- Listing the encryption status, see the **list ecp** command in "List" on page 33-20.
- Setting the device's name and encryption key, see the **set name** command in "Set" on page 33-23.

Monitoring Encryption

You can monitor the various encryption settings on the interfaces by:

1. Accessing the monitoring prompt using the **talk 5** command.
2. Selecting the interface you want to monitor using the **network x** command. This command puts you at the `PPP x>` prompt.

From this prompt, you can:

- List the current state of encryption, the most recent encryption negotiation, the elapsed time since an encryption state change, and the algorithms in use by the encrypters. (See the **list control ecp** command on page 34-3.)
- List the encryption control packets received and transmitted on the interface. (See the **list ecp** command on page 34-16.)
- List the encrypted data packets transmitted or received on the interface. (see the list edp command on page 34-17.)

Accessing the Interface Configuration Process

Use the following procedure to access the router's configuration process. This process gives you access to a specific interface's *configuration* process.

1. At the OPCON prompt (*), enter the **status** command to find the PID for CONFIG. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the OPCON **talk** command and the PID for CONFIG. (For more detail on this command, refer to Chapter 2, "The OPCON Process and Commands" on page 2-1.) For example:

```
* talk 6
```

After you enter the talk 6 command, the CONFIG prompt (Config>) displays on the console. If the prompt does not appear when you first enter **CONFIG**, press **Return** again.

3. At the CONFIG prompt, enter the **list devices** command to display the network interface numbers for which the router is currently configured. For example:

```
Config> list devices

Ifc 0 Ethernet                      CSR  81600, CSR2  80C00, vector 94
Ifc 1 WAN X.25                      CSR  81620, CSR2  80D00, vector 93
Ifc 2 WAN X.25                      CSR  81640, CSR2  80E00, vector 92
Ifc 3 WAN PPP                       CSR 381620, CSR2 380D00, vector 125
Ifc 4 WAN Frame Relay               CSR 381640, CSR2 380E00, vector 124
Ifc 5 Token Ring                    CSR 600000, vector 95
```

4. Record the interface numbers.
5. Enter the CONFIG **network** command and the number of the interface you want to configure. For example:

```
Config> network 1
```

The appropriate configuration prompt (such as TKR Config> for token-ring), now displays on the console.

Note: Not all network interfaces are user-configurable. For interfaces that cannot be configured, you receive the message:

```
That network is not configurable
```

Accessing the PPP Interface Configuration Prompt

To display the PPP config> prompt:

1. Enter **list devices** at the Config> prompt to display a list of interfaces.
2. If you have not already done so, set the data link protocol on one of the serial interfaces to PPP by entering **set data-link ppp** at the Config> prompt. For example:

```
Config> set data-link ppp
Interface Number [0]? 2
```

3. Enter **network** followed by the number of the PPP interface. For example:

```
Config> network 2
PPP config>
```

Point-to-Point Configuration Commands

Table 33-2 on page 33-17 summarizes the PPP configuration commands, and the rest of this section explains these commands. Enter the commands at the PPP config> prompt.

Table 33-2. Point-to-Point Configuration Command Summary

Command	Function
? (Help)	Displays all the Point-to-Point commands or lists the options for specific commands (if available).
Disable	Disables data compression (CCP), DTR line handling, CHAP, PAP, ECP. Also disables SPAP authentication in Remote LAN Access Features images.
Enable	Enables data compression (CCP), DTR line handling, CHAP, PAP, ECP. Also enables SPAP authentication in Remote LAN Access Features images.
List	Lists all information related to the point-to-point interfaces protocols, parameters, and options.
Set	Sets physical line (HDLC) parameters, LCP parameters, generic NCP parameters, and various NCP-specific options.
Exit	Returns to the `Config>` prompt.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
DISABLE
ENABLE
LIST
SET
EXIT
```

Example: **`list ?`**

```
ALL
BCP
CCP
HDLC
IPCP
LCP
NCP
```

Disable

Disables data compression, authentication protocols, multilink PPP, the Lower DTR feature, the DIALs feature, and SPAP authentication (SPAP authentication is supported *only* in DIALs Server images).

Syntax: disable ccp
chap
dials
ecp
lower-dtr
mp
pap
spap

ccp
: Disables the use of data compression on the interface. Refer to Chapter 19, “The Data Compression Subsystem” on page 19-1 for more information.

chap
: Disables the use of the Challenge-Handshake Authentication Protocol. Refer to “Challenge-Handshake Authentication Protocol (CHAP)” on page 33-9 for more information.

dials
: Disables the DIALs feature on this interface. Refer to Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1 for more information.

ecp
: This allows the router not to force the use of encryption on this interface. The interface will still accept and execute Encryption Control Protocol (ECP) if the peer is using ECP.

lower-dtr
: Determines the way the data terminal ready (DTR) signal is handled for leased serial-line interfaces that are disabled. If this parameter is set to “disabled” (the default) and the interface is disabled, the DTR signal is not dropped.

mp
: Disables the Multilink Protocol (MP) on this interface. See Chapter 35, “Using and Configuring the Multilink PPP Protocol” on page 35-1 for more information.

  Example: `disable mp`

  ```
  Disabled as a MP link
  ```

pap
: Disables the use of the Password Authentication Protocol. Refer to “Password Authentication Protocol (PAP)” on page 33-8 for more information.

spap
: Disables the use of the Shiva Password Authentication Protocol (SPAP).

  Note: SPAP is only available on interfaces that have IBM DIALs Dial-In circuits configured.

Enable

Enables data compression, encryption, authentication protocols, lower-DTR, multilink PPP protocol and the DIALs feature on this PPP interface. If multiple authentication protocols are enabled, the device attempts to use them in the following priority order:

1. SPAP
2. CHAP
3. PAP

Syntax: enable ccp
chap
dials
ecp
lower-dtr
mp

pap
spap

ccp
Enables the use of data compression on the interface. See Chapter 19, “The Data Compression Subsystem” on page 19-1 for more information.

chap
Enables the use of the Challenge-Handshake Authentication Protocol. You are prompted for a rechallenge interval. Specify 0 if you do not want to rechallenge periodically after the initial authentication phase is complete. Refer to “Challenge-Handshake Authentication Protocol (CHAP)” on page 33-9 for more information.

Syntax: enable chap

```
Rechallenge Interval in seconds (0=NONE)  [0] 10
CHAP enabled
```

dials
Enables the DIALs feature on this interface. Refer to Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1 for more information.

ecp
Enables the use of data encryption on this interface by negotiating Encryption Control Protocol (ECP). Once this is done, all PPP users with encryption enabled and with a valid encryption key must use ECP to connect to this port. PPP users without encryption enabled will still be able to connect to this interface.

lower-dtr
Determines the way the data terminal ready (DTR) signal is handled for leased serial-line interfaces that are disabled. If this parameter is set to “disabled” (the default) and the interface is disabled, the DTR signal is not dropped.

If Lower DTR is set to “enabled,” then the DTR signal will be dropped when the interface is disabled. This behavior may be desirable in situations where the interface has been configured as an alternate link for WAN Reroute and the interface is connected to a dial-out modem which maintains its dial connection based on the state of the DTR signal.

When the interface is disabled, the DTR signal is low and the modem keeps the dial connection down. When the interface is enabled, due to a WAN Reroute backup scenario, DTR is raised and the modem dials a stored number to the backup site. When the primary interface is restored, the alternate interface is disabled, DTR is lowered, and the modem hangs up the dial connection.

The following cable types are supported:

RS-232
V.35
V.36

Note: The **enable lower-dtr** command is not supported on PPP dial circuit interfaces.

mp

Enables the Multilink Protocol (MP) on this interface. See Chapter 35, “Using and Configuring the Multilink PPP Protocol” on page 35-1 for more information.

Example: `enable mp`

```
Enabled as a MP link
Is this link a dedicated MP link? [no] yes
MP interface for this MP link? [0] 3
```

pap

Enables the use of the Password Authentication Protocol. Refer to “Password Authentication Protocol (PAP)” on page 33-8 for more information.

spap

Enables the use of the Shiva Password Authentication Protocol (SPAP). Refer to “Shiva Password Authentication Protocol (SPAP)” on page 33-9 for more information. The **enable spap** command is available only in software loads with the DIALs feature.

List

Use the **list** command to display information related to the PPP interface and its protocol parameters and options.

Syntax: list
- all
- bcp
- ccp
- ecp
- hdlc
- ipcp
- lcp
- ncp

all

Lists all options and parameters related to the PPP interface.

The **list all** command displays the output of *all* the individual **list...** parameters described below.

bcp

Lists the Bridging Network control protocol options.

Example: **`list bcp`**

```
BCP Options
------------
Tinygram Compression:DISABLED
```

Tinygram Compression:

Displays whether Tinygram Compression is enabled/disabled.

ccp

Displays the currently selected data compression options. For additional information, see Chapter 19, “The Data Compression Subsystem” on page 19-1.

ecp

Displays the current Encryption Control Protocol state.

Example: **`list ecp`**

```
ECP Options
-----------
Data Encryption enabled
Algorithm list: DESE-CBC
DESE (Data Encryption Standard Encryption Protocol)
```

Data Encryption Enabled/Disabled
: Indicates whether data encryption is enabled or disabled on interface.

Algorithm List
: Displays the supported encryption algorithms. DES, as described by RFC 1969, is the only encryption algorithm currently supported.

hdlc
: Lists parameters related to the High-Level Data Link Control (HDLC) protocol. On PPP dial circuit interfaces, the “list hdlc” option is not available. For dial circuits, hardware data link parameters are a function of the base net rather than the PPP dial circuit. For additional information, see Chapter 49, “Configuring Dial Circuits” on page 49-1.

Note: This command is not supported on PPP dial circuit interfaces.

The Lower DTR state is displayed only if the cable type is not X.21.

Example: **`list hdlc`**

```
Encoding: NRZ
Idle State: Flag
Clocking:  Internal
Cable type:  V.35 DCE
Speed (bps): 6400

Transmit Delay Counter:  0
Lower DTR:  Disabled
```

Encoding:
: HDLC transmission encoding scheme, either NRZ (non-return to zero) or NRZI (non-return to zero inverted).

Idle State:
: Bit pattern, either Flag or Mark, transmitted on the point-to-point link when the interface is not transmitting data.

Clocking:
: Interface clocking, either external or internal.

Cable type:
: Specifies the type of cable in use (RS-232, V.35, or V.36).

Speed (bps):
: The physical data rate of the interface. When clocking is internal, this is the data rate generated by the internal clock.

Transmit Delay Counter:
: Number of flags sent between frames.

Lower DTR:
: Enabled or Disabled. If Lower DTR is enabled, the router drops the DTR signal when a WAN Reroute alternate link is no longer needed. Dropping the DTR signal causes the modem to terminate the leased-line connection for the alternate link.

ipcp
: Lists the Internet Protocol control protocol options.

Example: **`list ipcp`**

```
IPCP Options
------------
IPCP Compression:                None
Send Our IP Address:              Yes
Remote IP Address to Offer if Requested: 10.0.0.3
```

IPCP compression
: Indicates whether the PPP handler accepts compressed IP headers. PPP supports Van Jacobson TCP/IP header compression (RFC 1144). Enable this option when the point-to-point link is running at a low baud rate.

  A value of “Van Jacobson” indicates that header compression is supported. A value of “NONE” indicates that compressed headers are not being accepted.

Send Our IP Address
: Indicates where IPCP is configured to send the local IP address for this PPP interface to the remote end of the link in our initial “Configure Request.” Some PPP implementations require this information.

lcp
: Lists the parameters and options for the Link Control Protocol.

Example: **`list lcp`**

```
LCP Parameters
--------------
Config Request Tries:          20   Config Nak Tries:             10
Terminate Tries:               10   Retry Timer:                3000

LCP Options
-----------
Max Receive Unit:            2048   Magic Number:                 Yes
Peer to Local (Rx) ACCM:    A0000
Protocol Field Comp (PFC)      No   Addr/Cntl Field Comp(ACFC)    Yes

Authentication Options
---------------------- -------
Authenticate remote using:  none
Identify Self As              ibm
```

Config Request Tries:
: Number of times that LCP sends configure-request packets to a peer station while attempting to open a PPP link.

Config Nak Tries:
: Number of times that LCP sends configure-nak (“not acknowledged”) packets to a peer station while attempting to open a PPP link.

Terminate Tries:
: Number of times that LCP sends terminate-request packets to a peer station to close a PPP link.

Retry Timer:
: Number of milliseconds that elapse before packet transmission continues according to the number of times set by the “Config tries” parameter.

Max Receive Unit:
: Displays the maximum information field (packet) size handled by the link.

Peer to Local (Rx) ACCM
: Displays the characters that the peer must “escape” when transmitting packets to the router on asynchronous lines.

Magic Number:
Indicates whether the magic number loopback detection option is enabled.

Protocol Field Comp (PFC):
Indicates whether the PFC option is enabled.

Addr/Cntl Field Comp(ACFC):
Indicates whether ACFC is enabled.

Authenticate remote using:
A list of enabled authentication protocols.

Identify Self As:
The name set with the **set name** command.

ncp
Lists the parameters for all Network Control Protocols.

Example: **`list ncp`**

```
NCP Parameters
---------------
Config Request Tries:        20   Config Nak Tries:         10
Terminate Tries:             10   Retry Timer:              3000
```

Config Request Tries:
Number of times NCP sends configure-request packets to a peer station while attempting to open a PPP link.

Terminate Tries:
While awaiting a Terminate-Ack, the number of times NCP sends Terminate-Request before it closes a PPP link.

Config Nak Tries:
Number of times NCP sends configure-nak (not acknowledged) packets to a peer station while attempting to open a PPP link.

Retry Timer:
Number of milliseconds that elapse before timing out of NCP's transmission of configure-request packets (to open the link) and terminate-request packets (to close the link).

LLC

Use the **LLC** command to access the LLC configuration environment (available only if APPN is included in the software load). See “LLC Configuration Commands” on page 24-1 for an explanation of each of these commands.

Syntax: llc

Example: **`llc`**

```
LLC config>
```

Set

Use the **set** command to set HDLC parameters, LCP options and parameters, IPCP options, BCP options, and NCP parameters. “Parameters” are related to internal operations for such things as retry counts. “Options” are things that are negotiated with the other end.

Note: Values immediately following the command option prompts reflect the current setting of that option. They are not always the default values illustrated in this chapter.

Syntax: set

- bcp
- ccp options
- ccp algorithms
- hdlc...
- ipcp
- lcp...
- name
- ncp...

Note: The **set hdlc** commands are not supported on PPP dial circuit interfaces.

bcp

Sets the Bridging Control Protocol (BCP) parameters.

Example: **set bcp**

```
TINYGRAM COMPRESSION  [no]:
```

Tinygram Compression

Specifies whether or not Tinygram Compression is used. This option is useful for protocols that are prone to problems when bridged over low-speed (64 Kbps and below) lines. These protocols add zeroes between the data and the frame checksum to pad the Protocol Data Unit (PDU) to the minimum size. Tinygram compression removes the zeroes and preserves the frame checksum at the transmitting end. At the receiving end, it restores the packet to the minimum length.

ccp options

Prompts you for the configurable options of the compression algorithms. Some of the options may be modified later by PPP negotiations with the peer router on the WAN link. For additional information, see Chapter 19, “The Data Compression Subsystem” on page 19-1.

Example: **set ccp options**

```
STAC: # histories [1]?
STAC: check mode (0=none, 1-LCB, 2=CRC, 3=Seq) [3]?
```

STAC: # histories

This sets the number of compression “contexts” or “histories” that are used by the STAC compression engine.

A non-zero value means that the compression engine maintains the specified number of histories where it keeps information about previous data sent in packets. This historical data is used to improve the effectiveness of the compression.

The receiver maintains a similar history and as long as the transmitter and receiver keep their histories in sync, the receiver can properly decompress the packets it receives. If the histories get out of sync, packets are discarded as unusable data. Normally, you should set the number of histories to 1 unless the link quality is very poor.

A value of zero means that each packet sent is compressed without regard to any past packets sent and may always be reliably decompressed by the receiver. However, because the compressor cannot exploit any information derived from examining prior packets, the effectiveness of the compression usually is not as good.

Some implementations support more than one history, subdividing the data stream into separate streams that are compressed independently. The router does not support the use of more than one history on a PPP link.

STAC: check mode (0=none, 1=LCB, 2=CRC, 3=Seq)
STAC compressed datagrams normally include a check value used by the two ends of the link to recognize when a compressed packet has been lost or corrupted, and some action is needed to resynchronize the sender's and receiver's histories.

Note: Failure to detect a bad packet can cause all subsequent data to be decompressed incorrectly.

This option sets the exact form of check value used. Choose one of the following:

0 None: No check value is used. Without a check value, there is no way to determine that a packet has been lost, out-of-sequence, or corrupted. Do not use this mode unless the underlying data link provides reliable, sequenced packet delivery.

1 LCB: A "Longitudinal Control Byte" is used. This is a simple, 8-bit exclusive–OR checksum. *Its usage is strongly discouraged* because the receiver cannot detect a lost or an out-of-sequence packet, and the PPP frame checksum is a more reliable test of the packet's integrity.

2 CRC: A 16-bit cyclic redundancy checksum is used. Although this is a better test of a packet's integrity than the LCB, its use is still discouraged because the receiver still cannot use it to detect lost or out of sequence packets, and otherwise it becomes largely redundant with the frame checksum.

3 SEQ: An 8-bit sequence number is used (default). This is the preferred method of operation. If the number of histories is not 0, use of any other mode is strongly discouraged though another mode may be necessary for interoperability with certain non-RFC-compliant routers.

4 EXT: An extended mode that is similar to the sequence number mode, in that each packet includes a sequence number, but the compressed frame format is altered more radically. In extended mode, resynchronization with a peer is performed differently than with the other modes; the signaling between the two nodes is based upon flags passed in the headers of compressed datagrams rather than distinct CCP control packets.

Extended mode is provided for compatibility with certain non-RFC-compliant implementations. It should be used only with clients that do not support mode 3.

ccp algorithms ***list-of-algorithms***
Specifies an exact list of compression protocols to use. The order of preference depends on the order of entry in the list.

When the link negotiates compression with another node, it offers the entire list of protocols to the peer node in preference order. The peer node should select the first protocol it can use from the preference list. Enabling multiple protocols allows the peer to dictate which compression algorithm will be used on the link. If you need to avoid an algorithm, do not specify the algorithm in the list.

Specifying **none** disables the use of any protocol effectively disabling compression. The valid compression algorithms are:

STAC-LZS The STAC-LZS algorithm as described in RFC 1974

MPPC The Microsoft Point-to-Point Compression algorithm as described in RFC 2118.

Example: **`set ccp protocols`**

```
Enter a prioritized list of enabled compressors
(first is preferred), all on one single line.
Choices (can be abbreviated) are:
Stac-LZS, MPPC
Compressor list [Stac-LZS:]?
```

hdlc cable *cable type*

Set the HDLC cable type (that is connected to the interface) to one of the following types:

RS-232 DTE
RS-232 DCE
V35 DCE
V35 DTE
V36 DTE
X21 DCE
X21 DTE

A DTE cable is used when you are attaching the router to some type of DCE device (for example, a modem or a DSU/CSU).

A DCE cable is used when the router is acting as the DCE and providing the clocking for direct attachment.

hdlc clocking *external* or *internal*

To connect to a modem or DSU, configure clocking as external. To connect directly to another DTE device, use a DCE cable and set the clocking to “internal” at one end and to “external” at the other.

For internal clocking, you are prompted to enter a line speed in the range 2400 to 2048000, if you have not already set the line speed.

Example: **`set hdlc clocking internal`**

hdlc encoding *NRZ* or *NRZI*

Sets the HDLC transmission encoding scheme for an interface. Encoding may be set for NRZ (non-return to zero) or NRZI (non-return to zero inverted). NRZ is the more widely used encoding scheme while NRZI is used in some IBM configurations. The default value is NRZ.

Example: **`set hdlc encoding nrz`**

hdlc idle *flag* or *mark*

Sets the data link idle state to either Flag or Mark.

The flag option provides continuous flags (7E hex) between frames.

The mark option puts the line in a marking state (OFF, 1) between frames.

Example: **`set hdlc idle flag`**

hdlc speed *value*

For internal clocking, this command specifies the speed of the transmit and receive clock lines. The range is 2400 to 2 048 000 bps.

For external clocking, this command does not affect the hardware but it sets the speed some protocols, such as IPX, use to determine the routing parameters. In these cases, set the speed to match the actual line speed. If speed is not configured or is set to 0, the protocol assumes a speed of 1 000 000 bps. The maximum speed that can be configured if external clocking is used can be 6 312 000 bps.

Example: **`set hdlc speed 56000`**

hdlc transmit-delay *value*

Sets the number of flags sent between frames. The purpose of this command is to slow the serial line so that it is compatible with older, slower serial devices at the other end.

The range is 0 to 15. The default is 0.

Example: **`set hdlc transmit-delay 15`**

ipcp

Sets all Internet Protocol Control Protocol options for that link.

Example: **`set ipcp`**

```
IP COMPRESSION [yes]:
Number of Slots:   [16]?
Send our IP address [yes]:
Note: unnumbered interface addresses will not be sent.
Interface remote IP address to offer if requested (0 for none) [0.0.0.0]? 10.0.0.3
```

IPCP compression

Selects whether or not the PPP handler will accept compressed IP data. PPP supports Van Jacobson (VJ) TCP/IP header compression as described in RFC 1144. You should enable this option when the point-to-point link is running at a low baud rate.

Setting this value to yes enables the compression option. Setting this value to no disables the option. The default setting is no.

Slots

Sets the number IP headers that are saved for referential purposes when determining the type of compression that is enabled. The range is 1 to 16. The default is 16.

Send our IP address

Specifies whether or not to send the local IP address to the remote end of the link. You should set this option to “yes” if the other end of the link requires the IP address.

If set to “yes,” IPCP will send the IP address of the PPP interface, if the interface is configured with a numbered IP address, (That is, the address does not begin with 0). If this option is set to “no” and the peer sends us a Configure NAK with 0.0.0.0 for the IP Address option, the 2210 will respond with the address of the PPP interface if it is configured with a numbered address.

lcp *options* or *parameters*

Sets the Link Control Protocol options and parameters for the PPP link.

Example: **`set lcp options`**

```
Maximum Receive Unit (bytes) [2048]?
Magic Number [yes]:
Peer-to-Local Async Control Character Map (RX ACCM) [A0000] ?
Protocol Field Compression (PFC) [no]?
Addr/Cntl Field Compression (ACFC) [no]?
```

Maximum receive unit

Sets the maximum size of the information field that are transferred in a single datagram. The range is 576 to 4089 bytes. The default is 2048.

Magic number

Specifies whether or not the magic number option is enabled. The magic number provides a way of detecting looped back links in serial line configurations. When this option is enabled, the link uses the system clock as a random number generator. The random numbers that are generated are referred to as magic numbers.

When the LCP receives a Configure Request with a magic number present (i.e., the magic number option is enabled), the received magic number is compared with the magic number in the last Configure-Request sent to the peer. If the two magic numbers are different, the link is not considered looped back. If the two numbers are the same, the PPP handler attempts to bring the link down and up again to renegotiate magic numbers.

Setting this value to Yes enables the magic number option. Setting this value to No disables the option. The default setting is Yes.

Async Control Character Map

Indicates which characters that the peer must “escape” when transmitting packets to the router on asynchronous lines. This allows certain sensitive ASCII control characters, such as XON and XOFF, to be transmitted transparently over the link.

Specify a 32-bit bit mask in hexadecimal. If a bit in position ’N’ of the mask is set, the corresponding ASCII character ’N’ must be escaped (the LSB is bit number 0, corresponding to the ASCII NUL character).

The default value for this option is ’0A0000’, indicating that XON and XOFF (control-Q and control-S) need to be escaped. This is for the benefit of modems that use XON/XOFF to perform software handshaking. If this is not an issue, then it is recommended that you change the ACCM to zero (no characters escaped).

LCP is always willing to negotiate the ACCM, even on synchronous lines, and the **list lcp** command in the PPP monitoring process will display the negotiated value. However, synchronous lines employ a “bit-stuffing” mechanism rather than an “escaping” mechanism, so the ACCM is not normally meaningful on synchronous lines. It may be meaningful if the router is connected to a modem that performs sync-to-async conversion, in which case its value should reflect the requirements of the attached modem on the asynchronous side.

Addr/Cntl Field Compression (ACFC)

Specifies whether the peer can employ address and control field compression.

If the ACFC option is successfully negotiated by LCP, it means that the Address and Control field bytes which start off each packet may be omitted in the datagrams sent back and forth on the link. These bytes are always 0xFF 03, so there is no real information provided by them, and enabling ACFC means that the datagrams that are transmitted will be two bytes shorter.

To be precise, if you enable ACFC, you are indicating a receive-side capability. If you enable ACFC and LCP successfully negotiates it, the

other end can employ ACFC in the packets it transmits to the local end (most PPP options work like this). The local end will only transmit packets *without* the address and control fields if the other end also indicates its ability to handle such packets.

Enabling ACFC does not obligate the other end to send packets without the address and control fields, even if it accepts the option. Enabling ACFC merely tells the peer that it optionally *may* use ACFC, and the router will be able to handle the incoming packets. If the peer indicates that it can handle ACFC, then the router always performs ACFC on the packets it transmits regardless of whether ACFC is enabled locally.

LCP packets always are sent with address and control fields present. This guarantees that LCP packets will be recognized even if there is a loss of link synchronization.

Protocol Field Compression (PFC)
Specifies whether the peer is to employ protocol field compression.

When you specify “yes,” if the PFC option is negotiated successfully by LCP, the leading zero byte may be omitted from the “Protocol” field for those protocol values in the range ’0x0000’–’0x00FF’, for a one byte savings in the packets being transmitted. This range includes the majority of layer-3 protocol datagrams.

PPP protocol values are all assigned such that the upper byte of the protocol is an even value and the lower byte is an odd value (a limited use of the more generalized mechanism described by the ISO 3309 extension mechanism for address fields). Thus, the receiver can readily detect when the leading byte of a protocol value has been omitted (the first byte of the protocol field is odd rather than even), so there is no ambiguity interpreting frames in the presence of PFC.

PFC, like ACFC, is a receive side capability and the previous description of ACFC applies to PFC.

Example: **`set lcp parameters`**

```
Config tries  [20]?
NAK tries  [10]?
Terminate tries  [10]?
Retry timer (mSec)  [3000]?
```

Note: The value immediately following the command option prompt is the current setting of that option. It is not always the default value illustrated in this chapter.

Retry timer
Sets the amount of time in milliseconds that elapses before LCP's transmission of configure-request (to open the link) and terminate-request (to close the link) packets is timed out. Expiration of this timer causes a timeout and the halting of configure-request and terminate-request packet transmission. The range is 200 to 30000 milliseconds. The default setting is 3000 milliseconds.

Config tries
Sets the number of times that LCP sends configure-request packets to a peer station to establish the opening of a PPP link. The default value is 20. The range is 1 to 100.

The retry timer starts after the first configure-request packet is transmitted. This is done to guard against packet loss.

NAK tries
Sets the number of times that LCP sends configure-nak (nak = not acknowledged) packets to a peer station while attempting to open a PPP link. The default value is 10. The range is 1 to 100.

LCP sends configure-nak packets upon receiving configure-request packets with some unacceptable configuration options. These packets are sent to refuse the offered configuration options and to suggest modified, acceptable values.

Terminate tries
Sets the number of times that LCP sends terminate-request packets to a peer station to close a PPP link. The default value is 10. The range is 1 to 100.

The retry timer starts after the first terminate-request packet is transmitted. This is done to guard against packet loss.

name *routerid key*
Sets the name that the router uses when responding to authentication requests from another router. Also sets the device's encryption key.

Notes:

1. While the "case" you use for names and passwords sent to the peer on the link are preserved for this product, interoperability with other vendor products is easier if all names and passwords are entered in *lower* case.
2. Other implementations may not handle name and passwords with the same maximum length as supported in this product. The only indication would be a message from the authenticator stating that there is a bad name or password. If you receive this type of message, try shortening the routerid and key.

You will be prompted to enter the encryption key as 16 hexadecimal characters.

Example: **`set name routerid key`**

```
Config>
Config>net x
PPP x Config>
PPP x Config>set name
Enter Local Name:  []?newyork
Password:
Enter password again:
Enable encryption for this user/port (y/n) [No]:y
Encryption key should be 16 characters long.
Encryption Key (16 characters ) in Hex(0-9, a-f, A-F):
Encryption Key again (16 characters) in Hex(0-9, a-f, A-F):
PPP Local Name = newyork
PPP x Config>
```

ncp parameters
Sets the basic operational parameters for most NCPs.

Note: Although you access this command through a particular interface, this command will reset the parameters for all PPP interfaces.

Example:

```
set ncp parameters
```

```
Config tries  [20]
NAK tries  [10]?
Terminate tries  [10]?
Retry timer (mSec)  [3000]?
```

Config tries — Sets the number of configure-request packets sent by NCP to a peer station to attempt to open a PPP link. The range is 1 to 100. The default is 20.

This action indicates the desire to open an NCP connection with a specified set of configuration options. The retry timer starts after a configure-request packet is transmitted. This is done to guard against packet loss.

NAK tries — Sets the number of configure-nak (nak = not acknowledged) packets that NCP sends to a peer station while attempting to open a PPP link. The range is 1 to 100. The default value is 10.

Upon receiving configure-request packets with some unacceptable configuration options, NCP sends configure-nak packets. These packets are sent to refuse the offered configuration options and to suggest modified, acceptable values.

Terminate tries — Sets the number of terminate-request packets sent by NCP to a peer station to close a PPP link. The range is 1 to 100. The default value is 10.

This action indicates the desire to close an NCP connection. The retry timer is started after a terminate-request packet is transmitted. This is done to guard against packet loss.

Retry timer — Sets the amount of time, in milliseconds, that elapses before NCP's transmission of configure-request (to open the link) and terminate-request (to close the link) packets is timed out. Expiration of this timer causes a timeout and the halting of configure-request and terminate-request packet transmission. The range is 200 to 30000 milliseconds. The default is 3000 milliseconds.

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

Chapter 34. Monitoring Point-to-Point Protocol Interfaces

This chapter describes how to monitor specific Point-to-Point Protocol interfaces in the router. Sections in this chapter include:

Accessing the Interface Console Process

To access the PPP interface console process, do the following:

1. Enter **interface** at the + prompt to display a list of configured interfaces.
2. Enter **network** followed by the number of the PPP interface.

```
+ network 2
PPP>
```

Point-to-Point Console Commands

This section summarizes and then explains the Point-to-Point console commands. Enter the commands at the PPP> prompt. Table 34-1 shows the commands.

Note: The options available for these commands depend on what protocols are available in the router software. For example, when the router software (image) does not contain APPN support, the **list isrcp**, **list isr**, **list hprcp**, **list hpr**, and **llc** commands are not available.

Table 34-1. Point-to-Point Console Command Summary

Command	Function
? (Help)	Displays all the Point-to-Point commands or lists subcommand options for specific commands (if available).
Clear	Clears all statistics from point-to-point interfaces.
List	Displays information and counters related to the point-to-point interface and PPP parameters and options.
LLC	Displays the LLC console monitoring prompt.
Exit	Exits the Point-to-Point console process.

? (Help)

Use the **? (Help)** command to obtain a list of the commands available from that prompt level. You can also enter this command after specific command names to obtain a listing of the command options available for that command.

Syntax: ?

Example: **?**

```
CLEAR
LIST
LLC
EXIT
```

Example: `list ?`

Clear

Use the **clear** command to clear all statistics from point-to-point interfaces.

Syntax: clear

Example: `clear`

List

Use the **list** command to display information and counters related to the point-to-point interface and PPP parameters and options.

Syntax:; list

all	
control	
errors	
interface	
lcp	- PPP link CP
pap	- PAP Authentication CP
chap	- CHAP Authentication CP
ecp	- Encryption Control Protocol
edp	- Encrypted packet statistics
spap	- SPAP Authentication CP
ccp	- PPP Compression CP
cdp	- PPP compression
compression	- PPP compression
bcp	- Bridging (ASRT) CP
brg	- Bridging (ASRT)
stp	- Spanning Tree Protocol
nbcp	- Netbios
nbfcp	- Netbios Frame
ipcp	- Internet Protocol CP
ip	- Internet Protocol
ipxcp	- Novell IPX CP
ipx	- Novell IPX
atcp	- Appletalk (Phase 2) CP
ap2	- Appletalk (Phase 2)
dncp	- DECnet IV CP
dn	- DECnet IV
osicp	- ISO's OSI CP
osi	- ISO's OSI
bvcp	- Banyan VINES CP
vines	- Banyan VINES
isrcp	- APPN ISR CP
isr	- APPN ISR
hprcp	- APPN HPR CP
hpr	- APPN HPR

all

Lists all information and counters related to the point-to-point interface and PPP options and parameters. The output displayed for this command is a combination of the displays from all of the individual **list** *item* commands.

Note: If a network control protocol is not available on an interface, a message is displayed indicating that no protocol or statistics information is available for that network control protocol's list commands.

Example: `list all`

control
: Lists negotiated options or other state information for a control protocol.

  ccp
  ecp
  lcp
  bcp
  nbcp
  nbfcp
  ipcp
  ipxcp
  atcp
  dncp
  osicp
  bvcp
  isrcp
  hprcp

Example: `list control ccp`

```
CCP State:                Open
Previous State:           Ack Sent
Time Since Change:        264 hours, 56 minutes and 58 seconds

Compressor:   STAC-LZS histories 1, check_mode SEQ
Decompressor: STAC-LZS histories 1, check_mode SEQ

Max size of compression dictionary:   12494.
Max size of decompression dictionary: 4424.
```

CCP state
: The current state of the point-to-point link. If “Open” then compression was successfully negotiated on this link. If not open, compression is not running on the link.

Previous State
: State of the point-to-point link before the state displayed in the current state field.

Compressor
: Shows which compressor was negotiated and the options it is using.

Decompressor
: Shows which decompressor was negotiated and the options it is using.

Max size of compression dictionary
: The size of the data space allocated for the compression “context” or “history.”

Max size of decompression dictionary
: The size of the data space allocated for the decompression “context” or “history.”

Example:

```
PPP x>list control ecp

ECP State:                Open
Previous State:           Ack Sent
Time Since Change:        16 minutes and 40 seconds

Local (transmit) encrypter: DES
Remote (receive) encrypter: DES
```

ECP State:
The current state of the point-to-point link. If “Open” then encryption was successfully negotiated on this link. If not “Open,” encryption is not running on the link.

Previous State:
The state of the point-to-point link before the state displayed in the current state field.

Time Since Change:
The elapsed time between the above two state changes.

Local (transmit) encrypter:
This encryption algorithm is used for encrypting the data being sent on this PPP interface.

Remote (receive) encrypter:
The encryption algorithm is used for decrypting the received data on this interface.

Example: `list control lcp`

```
Version:                    1
            Link phase:                 Establishing connection (LCP)
            LCP State:                  Listen
            Previous State:             Req Sent
            Time Since Change:          1 minute and 57 seconds
            Remote Username:            - No Authentication -
            Last Identification Rx'd
            Time Connected:             - No Connection -

            LCP Option                  Local                       Remote
            ----------                  -----                       ------
            Max Receive Unit:           2048                        1500
            Async Char Mask:            FFFFFFFF                    FFFFFFFF
            Authentication:             None                        None
            Magic Number:               7A8CBFD7                    None
            Protocol Field Comp:        No                          No
            Addr/Cntl Field Comp:       No                          No
            32-Bit Checksum:            No                          No
```

Version
Displays the current version of the Point-to-Point Protocol.

Link phase
Displays the current activity on the link. This can have one of the following values:

Dead There is no activity on the link; the interface is down.

LCP The link is in LCP negotiation. This state occurs when first bringing up an interface. The interface may be in self-test at this time.

Authenticate The link is performing initial authentication.

ECP The link is negotiating an encryption algorithm.

Ready Link is operating normally. NCPs can negotiate and data traffic associated with can flow after successful NCP negotiation.

Terminate The link is being shut down.

LCP State
: Displays the current state of the point-to-point link. These states include the following:

  OPEN - Indicates that a connection has been made and data can be sent. The retry timer does not run in this state.

  CLOSED - Indicates that the link is down and no attempt is being made to open it. In this state, all connection requests from peers are rejected.

  LISTEN - Indicates that the link is down and no attempt is being made to open it. In contrast to the CLOSED state, however, all connection requests from peers are accepted.

  REQUEST-SENT - Indicates that an active attempt is being made to open the link. A Configure-request packet has been sent but a Configure-Ack has not yet been received nor has one been sent. The retry timer is running at this time.

  ACK-RECEIVED - Indicates that a Configure-request packet has been sent and a Configure-Ack packet has been received. The retry timer is still running since a Configure-Ack packet has not been transmitted.

  ACK-SENT - Indicates that a Configure-Ack packet and a Configure-request packet have been sent but a Configure-Ack packet has not been received. The retry timer always runs in this state.

  CLOSING - Indicates that an attempt is being made to close the connection. A Terminate-request packet has been sent but a Terminate-Ack packet has not been received. The retry timer is running in this state.

Previous State
: Displays the state of the point-to-point link prior to the state displayed in the Current state field. These states are the same as those described in the Current state field.

Time since change
: Displays the amount of time that has elapsed since the last link state change.

Remote Username
: When authentication is required on the link, this field shows the name that the peer supplied.

Last Identification Rx'd
: An optional packet type that is defined for LCP is an “Identification” packet. The contents of this packet are undefined but are normally expected to be a human-readable string provided by the peer to give some identifying information such as a name, manufacturer, model number, or other information the manufacturer wishes to provide. If the router receives such a packet, the contents of the last such packet received are displayed here.

Time Connected
: Indicates how long the peer has been connected on this link.

LCP Option
: These fields indicate the values of options that have been negotiated with the peer when LCP is in the Open state. When LCP is not open, these values represent initial defaults or configured values that will be used in subsequent LCP negotiations.

Max Receive Unit
: Indicates the maximum length for the packet size that the local and remote ends can transmit. This is the maximum length of the payload portion of a PPP packet and it does not include PPP header and trailer bytes.

  When LCP is in an Open state, the values indicate the lengths that have been negotiated with the peer. The router does not support differing MRU lengths for the peer and local end, so these values will be the same.

Async Character Mask
: This indicates the asynchronous control character mask that has been negotiated. The router accepts ACCM negotiation even on synchronous lines, although this does not affect the actual packet data sent. See the **set lcp options** command on page 33-27 for more information about the ACCM.

Authentication
: Indicates which authentication protocol, if any, each end of the link requires. Multiple protocols may be available at each end; this value indicates which protocol the units agreed to use.

Magic number
: Displays the current magic number being used for both the local and remote ends of the link for loopback detection.

Protocol compression
: Indicates whether PFC has been negotiated.

Address/Control compression
: Indicates whether ACFC has been negotiated.

32-bit checksum
: Not currently supported. PPP will reject this option if it is received.

Example: `list control bcp`

```
BCP State:            Closed
Previous State:        Closed
Time Since Change:     5 hours, 25 minutes and 3 seconds

BCP Option             Local                    Remote
Tinygram Compression   DISABLED                 DISABLED
Source-route Info:
Remote side does not support source-route bridging
```

The BCP State fields are the same as those described under the **list control lcp** command.

Tinygram Compression
: Displays whether or not Tinygram Compression is enabled or disabled on the local and remote ends of the link.

Source-route Info
: Displays whether or not source route bridging is enabled for the local and remote ports that correspond to this interface.

Example: `list control nbcp`

```
NBCP State:               Closed
Previous State:           Closed
Time Since Change:        3 hours, 48 minutes and 24 seconds

NetBIOS Control Protocol Info:
Local  MAC Address = 0x000000000000
Remote MAC Address = 0x000000000000
Remote NetBIOS Names: (0)
```

The NBCP State fields are the same as those described under the **list control lcp** command.

Local MAC Address
The Local MAC Address is the MAC Address that is used by the DOS/Win DIALs client. It is a pseudo-random number, or a Locally Administered Address (LAA), if you configured an LAA in the client.

Remote MAC Address
The Remote MAC Address is the MAC Address that the 2210 DIALs Server has assigned to this client for use on the LAN.

Remote NetBIOS Name
The list of NetBIOS names of LAN resources to which the client has requested access.

Example: `list control nbfcp`

```
NBFCP State:              Closed
Previous State:           Closed
Time Since Change:        4 hours, 5 minutes and 58 seconds

NetBIOS Frame Control Protocol Info:
Local  MAC Address = 0x000000000000
Remote MAC Address = 0x444553540000
Remote NetBIOS Names: (0)

Remote Peer Class:        0
Remote Peer Version Major: 0
Remote Peer Version Minor: 0
```

The NBFCP State fields are the same as those described under the **list control lcp** command.

Local MAC Address
The Local MAC Address is the MAC Address that is used by the Win 95/NT Dial-Up Networking client. It is a pseudo-random number, or a Locally Administered Address (LAA), if you configured an LAA in the client.

Remote MAC Address
The Remote MAC Address is the MAC Address that the 2210 DIALs Server has assigned to this client for use on the LAN.

Remote NetBIOS Name
The list of NetBIOS names of LAN resources to which the client has requested access.

Remote Peer
The Remote Peer Class, Version Major, and Version Minor is the information passed back to the 2210 by the NBFCP Peer Information option.

Example: `list control ipcp`

```
IPCP State:               Listen
Previous State:           Closed
Time Since Change:        1 hour, 57 minutes and 52 seconds

IPCP Option               Local               Remote
-----------               -----               ------
IP Address                0.0.0.0             10.0.0.152
Compression Slots         None                None


DHCP State:               BOUND
Lease Server:             10.0.0.111
Leased IP Address:        10.0.0.152
Lease Time:               4 minutes and 0 seconds
Renewal Time:             2 minutes and 0 seconds
Rebind Time:              3 minutes and 30 seconds
Lease Time Elapsed:       1 second
Lease Time Remaining:     3 minutes and 59 seconds

DHCP Client ID:           0100120B0000
```

The IPCP state fields are the same as those described under the **list control lcp** command.

IP Address:
: Indicates if this interface's IP address (Local) and the negotiated address of the remote (Remote), if any.

Compression Slots
: Indicates the number of IP headers saved for referential purposes when determining the type of compression that is enabled.

DHCP State
: This is the Proxy DHCP as described in RFC 1541.

Lease Server
: The server from which the lease was acquired.

Leased IP address
: The address leased to the client. This address should be equivalent to the "Remote IP Address" listed above.

Lease Time
: Length of lease from the DHCP server for this address. When "Lease Time Elapsed" equals this time, the lease will be expire and the IPCP connection closed.

Renewal Time
: Time after which Proxy DHCP attempts to extend this lease from the server. When "Lease Elapsed Time" equals this time, Proxy DHCP attempts to renew the lease, resetting the "Lease Time," "Lease Elapsed Time," and "Lease Time Remaining," if successful.

Rebind Time
: Time before Proxy DHCP attempts to obtain a new lease from any configured DHCP server. When "Lease Elapsed Time" equals this time, Proxy DHCP attempts to obtain a new lease, resetting the "Lease Time," "Lease Elapsed Time," and "Lease Time Remaining," if successful.

Leased Time Elapsed
: Time elapsed for this lease. This is not necessarily the time for this particular dial-in session, as the lease may have been renewed. When the lease is renewed, this timer is set back to 0.

Leased Time Remaining
: Time remaining for this lease. This parameter is equal to “Lease Time” minus “Lease Time Elapsed.”

DHCP client ID
: A unique ID for this client (dial-in user). All DHCP messages are identified to and from the DHCP server by this client ID.

Example: `list control ipxcp`

```
IPXCP State:          Closed
Previous State:       Closed
Time Since Change:    2 hours, 9 minutes and 9 seconds
```

The IPXCP state fields are the same as those described under the **list control lcp** command.

Example: `list control atcp`

```
ATCP State:           Closed
Previous State:       Closed
Time Since Change:    6 hours, 27 minutes and 7 seconds

AppleTalk Address Info:
Common network number = 12
Local node ID = 49
Remote node ID = 76
```

The ATCP State fields are the same as those described under the **list control lcp** command.

Common Network Number
: Network number of the two ends of the point-to-point link. (You must statically configure both ends of the link to have the same network number.)

Local Node ID
: Unique node number of the local end of the link.

Remote Node ID
: Unique node number of the remote end of the link.

Example: `list control dncp`

```
DNCP State:               Closed
Previous State:           Closed
Time Since Change:        2 hours, 2 minutes and 58 seconds
```

The DNCP state fields are the same as those described under the **list control lcp** command.

Example: `list control osicp`

```
OSICP State:         Closed
Previous State:      Closed
Time Since Change:   6 hours, 28 minutes and 32 seconds
```

The OSICP State fields are the same as those described under the **list control lcp** command.

Example: `list control bvcp`

```
BVCP State:               Open
Previous State:           Ack Sent
Time Since Change:        403 hours, 49 minutes and 2 seconds
```

The BVCP State fields are the same as those described under the **list control lcp** command.

Note: The command word **bvcp** and the acronym BVCP stand for the Banyan VINES Control Protocol (BVCP).

Example: `list control isrcp`

```
APPN ISRCP State:          Open
Previous State:            Ack Rcvd
Time Since Change:         1 hour, 48 minutes and 5 seconds
```

The APPN ISR control protocol (ISRCP) state fields are the same as those described under the list control lcp command.

Example: `list control hprcp`

```
APPN HPRCP State:          Open
Previous State:            Ack Rcvd
Time Since Change:         1 hour, 48 minutes and 10 seconds
```

The APPN HPR control protocol (HPRCP) state fields are the same as those described under the list control lcp command

error

Lists information related to all error conditions tracked by the PPP software.

Example: `list error`

```
Error Type                 Count          Last One
----------                 -----          --------
Bad Address:                  0               0
Bad Control:                  0               0
Unknown Protocol:             0               0
Invalid Protocol:             0               0
Config Timeouts:              0               0
Terminate Timeouts:           0               0
```

Bad address

Indicates the total number of bad addresses encountered over the point-to-point link. “Bad addresses” refers to the HDLC framing byte at the start of the packet.

Bad control

Indicates the total number of bad control packets encountered over the point-to-point link. “Bad control” refers to the 0x03 prefix on HDLC encapsulated PPP packets (“UI” value that follows the 0xFF).

Unknown protocol

Indicates the total number of unknown protocol packets encountered by the current link.

Invalid protocol

Indicates the total number of invalid protocol packets encountered by the current link.

Config timeouts

Indicates the total number of configuration timeouts experienced by the link.

Terminate timeouts

Indicates the total number of link termination timeouts experienced by the link.

interface

Lists PPP interface statistics.

Example: `list interface`

```
Interface Statistic          In              Out
------------------           --              ---
Packets:                      0               0
Octets:                       0               0
```

Packets
: Indicates the number of packets received and transmitted on this interface.

Octets
: Indicates the number of octets received and transmitted on this interface.

lcp
: Lists statistics for the Link Control Protocol.

Example: `list lcp`

```
LCP STATISTIC        IN       OUT
-------------        --       ---
PACKETS:             42       42
OCTETS:              1260     1260
CFG REQ:             0        0
CFG ACK:             0        0
CFG NAK:             0        0
CFG REJ:             0        0
TERM REQ             0        0
TERM ACK             0        0
ECHO REQ:            21       21
ECHO RESP:           21       21
DISC REQ:            0        0
CODE REJ:            0        0
```

Packets
: Indicates the total number of LCP packets transmitted (out) and received (in) over the current point-to-point interface.

Octets
: For LCP frames, indicates the total number of bytes in octets transmitted and received over the current point-to-point interface.

CFG REQ
: Indicates the total number of configure-request LCP packets transmitted and received over the current point-to-point interface.

CFG ACK
: Indicates the total number of configure-ack (acknowledged) LCP packets transmitted and received over the current point-to-point interface.

CFG NAK
: Indicates the total number of configure-nak (not acknowledged) LCP packets transmitted and received over the current point-to-point interface.

CFG REJ
: Indicates the total number of configure-reject LCP packets transmitted and received over the current point-to-point interface.

TERM REQ
: Total number of terminal request LCP packets transmitted and received over the current point-to-point interface.

TERM ACK
: Total number of terminal ack LCP packets transmitted and received over the current point-to-point interface.

ECHO REQ
: Indicates the total number of echo-request LCP packets transmitted and received over the current point-to-point interface.

ECHO RESP
: Indicates the total number of echo-response LCP packets transmitted and received over the current point-to-point interface.

DISC REQ
: Indicates the total number of discard-request LCP packets transmitted and received over the current point-to-point interface.

CODE REJ
: Indicates the total number of code-reject LCP packets transmitted and received over the current point-to-point interface.

pap
: Lists statistics for the Password Authentication Protocol.

Example: `list pap`

```
PAP Statistics            In                    Out
--------------            --                    ---
Packets:                  0                     0
Octets:                   0                     0
Requests:                 0                     0
Acks:                     0                     0
Naks:                     0                     0
```

Packets
: The total number of PAP packets sent or received.

Octets
: The number of bytes of data that were sent or received in those packets.

Requests
: The number of PAP “Request” packets sent or received. These are the packets which contain the PAP name/password pairs.

Acks
: The number of Acks (success replies) sent or received for the PAP requests (for example, if the peer sends a valid Request packet, the router replies with an Ack).

Naks
: The number of Naks sent or received for the PAP requests (for example, if the peer sends an invalid Request packet, the router replies with a Nak).

chap
: Lists statistics for the Challenge-Handshake Authentication Protocol.

Example: `list chap`

```
CHAP Statistics           In                    Out
---------------           --                    ---
Packets:                  0                     0
Octets:                   0                     0
Challenges:               0                     0
Responses:                0                     0
Successes:                0                     0
Failures:                 0                     0
```

Packets
: The total number of CHAP packets sent or received.

Octets
: The number of bytes of data that were sent or received in the packets.

Challenges
: The number of CHAP “Challenge” packets sent or received. A CHAP Challenge packet includes a randomly generated encryption key and is a demand on the peer to generate a suitable response based on that key and on stored password information.

Responses
: The number of CHAP “Response” packets sent or received. A Response packet contains a peer's answer to a “Challenge” request.

Successes/Failures
: The number of Success or Failure packets sent or received. A unit sends out a Challenge packet and waits for the peer's Response reply. It then examines the Response packet and sends a Success or Failure packet to indicate whether the Response was valid.

  These counters reflect the number of Success or Failure packets sent. A peer gets several tries to respond successfully before authentication is considered to have failed.

spap
: Lists statistics for the Shiva Password Authentication Protocol.

  Example: `list spap`

```
SPAP Statistic            In                    Out
--------------            --                    ---
Packets:                  0                     0
Octets:                   0                     0
Requests:                 0                     0
Acks:                     0                     0
Naks:                     0                     0
Dialbacks:                0                     0
PleaseAuthenticates:      0                     0
Change Passwords:         0                     0
Alerts:                   0                     0
```

  Packets
  : The total number of SPAP packets sent or received.

  Octets
  : The number of bytes of data that were sent or received in those packets.

  Requests
  : The number of SPAP “Request” packets sent or received. These are the packets which contain the SPAP name/password pairs.

  Acks
  : The number of Acks (success replies) sent or received for the SPAP requests (for example, if the peer sends a valid Request packet, the router replies with an Ack).

  Naks
  : The number of Naks sent or received for the SPAP requests (for example, if the peer sends an invalid Request packet, the router replies with a Nak).

  Dialbacks
  : The number of times a user:

    - Requested a call-back (roaming callback) and it was granted.
    - Dialed-in and they were configured for required callback and dialed back at the predetermined number stored in the user profile.

PleaseAuthenticates
: The number of SPAP please authenticate packets that have been sent or received on this interface. An SPAP please authenticate packet is sent as the result of a timeout when waiting for the other end to send an SPAP authenticate request.

Change Passwords
: The number of change password requests that sent or received on this interface.

Alerts
: The number of SPAP banners that have been sent or received.

ccp
: Lists statistics for compression control protocol.

Example: `list ccp`

```
CCP   Statistic           In                    Out
---------------           --                    ---
Packets:                  24                    25
Octets:                   174                   177
Reset Reqs                0                     0
Reset Acks:               0                     0
Prot Rejects:             0                     0
```

Packets
: Indicates the number of packets received and transmitted on this interface.

Octets
: Indicates the number of octets received and transmitted on this interface.

Reset Reqs
: The number of CCP dictionary “Reset Requests” that were transmitted or received.

Reset Acks
: The number of CCP dictionary “Reset Acknowledgments” that were transmitted or received.

  Reset Request and Reset Acknowledgment packets are control packets passed between the CCP entities at each end, used to maintain synchronization of the data dictionaries at each end of the link.

Prot Rejects
: Indicates the number of protocol rejects of CCP packets sent by the peer (reception of a protocol reject would signify that the peer does not support CCP).

cdp
: Displays statistics associated with compressed data packets sent or received on this interface.

Example: `list cdp`

```
Compression Statistic      In                      Out
---------------------      --                      ---
Packets:                   31035                   46550
Octets:                    1614885                 2421137
Compressed Octets:         931416                  1521039
Incompressible Packets:    0                       0
Discarded Packets:         0                       0
Copied Packets:            1                       0
Prot Rejects:              0                       -

Compressor (transmit) statistics:
    Recent compression ratio: 1.7:1
Decompressor (receive) statistics:
    Recent compression ratio: 1.7:1
```

Packets

These counters indicate the number of compressed datagrams sent and received. On the output side, the count includes only those packets that were actually sent as PPP compressed datagrams; it does not include packets that were found to be incompressible and sent in their original uncompressed form.

These counters count the packets sent or received that had the PPP protocol type of X'00FD' (CDP). When STAC extended mode or MPPC has been negotiated, incompressible packets may be encapsulated in CDP datagrams. This encapsulation would include the incompressible packets in these counts.

Octets

These counters indicate the number of bytes effectively transmitted or received in compressed form. These counts reflect the lengths of the original datagrams before compression or after decompression.

Compressed octets

These counters indicate the number of bytes for all of the compressed datagrams sent and received. These counts are the lengths of the actual CDP packets after compression or before decompression.

Incompressible packets

These counters indicate the number of packets that were incompressible and therefore sent in original uncompressed form.

Discarded packets

These counters indicate how many packets were discarded because they could not be successfully decompressed. Typically these packets will be packets that the peer was transmitting just after the router has sent a Reset-Request, but before the peer has received and processed the Reset-Request. Packets are also dropped if the router detects that data in the packets is incorrect. An example of incorrect data is a packet that contains a bad sequence number.

If the number of discarded packets increases too rapidly, then packets are being lost or corrupted on the line, probably due to noise on the line, and the link performance may be degraded.

Protocol rejects

This counter indicates the number of Protocol-Rejects of CDP packets that have been received from a peer. This count should be zero, because the link will not send CDP packets if the use of compression has not already been negotiated.

Compression ratios

The ratios give an approximate indication of the effectiveness of the compressor and decompressor. These ratios are based on the number of plain-text bytes divided by the number of corresponding compressed bytes, so values greater than 1 are preferable for both input and output. The higher the number, the more effective the compression.

The output ratio is computed as the ratio of the number of original plain-text bytes divided by the number of bytes sent as a result of attempting compression - whether the packet actually was compressed or sent as a CDP packet. If a data stream does not compress well and most of the packets are sent in their original form or in enlarged CDP packets, the compression output ratio will drop. If the ratio drops below 1.0, the compressor is actually reducing the effective bandwidth of the line rather than increasing it, and should be disabled on that interface if the state persists for a long time.

The input ratio is computed based on the number of bytes received in CDP frames divided into the number of decompressed bytes. Unlike the output ratio, this count does not include any packets that were incompressible and sent in plain-text form. This is because the router cannot determine if a received non-CDP packet was an incompressible packet that the peer sent in plain-text form, or just a packet that the peer did not attempt to compress.

Because of the method of calculation, the output ratio on one end of the link does not necessarily match the input ratio at the other end.

compression

This command displays the same information as list cdp.

ecp

Lists statistics for encryption control protocol packets sent or received on the interface.

Example:

```
PPP x>list ecp

ECP Statistic              In                      Out
-------------              --                      ---
Packets:                   2                       2
Octets:                    26                      26
Reset Reqs:                0                       0
Reset Acks:                0                       0
Prot Rejects:              0                       -
Local (transmit) encrypter: DES
Remote (receive) encrypter: DES
```

Packets

Indicates the total number of ECP packets transmitted (out) and received (in) over the current point-to-point interface.

Octets

Indicates the total number of bytes transmitted and received in the ECP packets.

Reset Reqs

Indicates the number of Reset requests transmitted and received on this interface. A Reset Request will be sent whenever ECP discard an EDP packet.

Note: Because DES, the only supported encryption algorithm, does not send reset requests this number will be zero.

Reset Acks
: Indicates the reset acknowledgments transmitted and received on this interface. A Reset Ack packet will be sent for every Reset Request packet received.

  Note: Because DES, the only supported encryption algorithm, does not send any Reset Requests this number will be zero.

Prot Rejects
: Indicates the total number of protocol reject packets transmitted and received over the current point-to-point interface.

Local (transmit) encrypter
: This encryption algorithm will be used to encrypt the data being sent on this point-to-point interface.

Remote (receive) encrypter
: This encryption algorithm will be used to decrypt the received data on this point-to-point interface.

edp
: Lists statistics associated with the encrypted packets being sent or received on the interface.

  Example:

```
PPP x>list edp

Encryption Statistic       In                      Out
---------------------      --                      ---
Packets:                   20                      30
Octets:                    29164                   44790
Encrypted Octets:          29280                   44880
Discarded Packets:         0                       0
Prot Rejects:              0                       -
```

Packets
: Indicates the total number of IP packets transmitted (out) and received (in) over the current point-to-point interface.

Octets
: Indicates the total number of octets of data bytes transmitted and received over the current IP connection.

Encrypted Octets
: Indicates the number of encrypted octets transmitted or received on this interface.

Discarded Packets
: Indicates the number of packets that were discarded because they could not be successfully decrypted.

Prot Rejects
: Indicates the total number of protocol reject packets transmitted and received over the current point-to-point interface.

bcp
: Lists statistics for the Bridging control protocol. These fields are the same as those described under the **list ip** command. (See 34-19.)

  Example: `list bcp`

```
BCP Statistic           In                  Out
--------------          --                  ---
Packets:                0                   0
Octets:                 0                   0
Prot Rejects:           0                   -
```

brg

Lists statistics on the bridge packets received and transmitted over the PPP interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list brg`

```
BRG Statistic           In                  Out
--------------          --                  ---
Packets:                0                   0
Octets:                 0                   0
Prot Rejects:           0                   -
```

stp

Lists statistics for the spanning tree protocol. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list stp`

```
Spanning Tree Statistic       In            Out
-----------------------       --            ---
Packets:                      0             0
Octets:                       0             0
```

nbcp

Lists NetBIOS Control Protocol statistics for the point-to-point interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: list nbcp

```
NBCP Statistic            In                  Out
---------------           --                  ---
Packets:                  0                   0
Octets:                   0                   0
Prot Rejects:             0                   -
```

nbfcp

Lists NetBIOS Frame Control Protocol statistics for the point-to-point interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: list nbfcp

```
NBFCP Statistic           In                  Out
---------------           --                  ---
Packets:                  0                   0
Octets:                   0                   0
Prot Rejects:             0                   -
```

ipcp

Lists Internet Protocol Control Protocol statistics for the point-to-point interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list ipcp`

```
IPCP STATISTIC       IN         OUT
--------------       --         ---
PACKETS:             0           0
OCTETS:              0           0
PROT REJECTS:        0
```

ip

Lists all information related to IP packets over the point-to-point link.

Example: `list ip`

```
IP Statistic         In         Out
------------         --         ---
Packets:             349        351
Octets:              128488     129412
Prot Rejects:        0          -
```

Packets

Indicates the total number of IP packets transmitted (out) and received (in) over the current point-to-point interface.

Octets

Indicates the total number of octets transmitted and received over the current IP connection.

Prot Rejects

Indicates the total number of protocol reject packets transmitted and received over the current point-to-point interface.

ipxcp

Lists statistics for the IPX control protocol. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list ipxcp`

```
IPXCP Statistic           In               Out
---------------           --               ---
Packets:                  0                0
Octets:                   0                0
Prot Rejects:             0                -
```

ipx

Lists IPX statistics for the point-to-point interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list ipx`

```
IPX Statistic           In        Out
------------            --         ---
Packets:                0          0
Octets:                 0          0
Prot Rejects:           0          -
```

atcp

Lists statistics for the AppleTalk control protocol. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list atcp`

```
ATCP Statistic           In                    Out
---------------          --                    ---
Packets:                 0                     0
Octets:                  0                     0
Prot Rejects:            0                     -
```

ap2

Lists AppleTalk Phase 2 statistics for the point-to-point interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list ap2`

```
AP2 Statistic        In         Out
------------         --         ---
Packets:             349        351
Octets:              128488     129412
Prot Rejects:        0
```

dncp

Lists statistics on the DECnet control protocol packets. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list dncp`

```
DNCP Statistic          In                Out
--------------          --                ---
Packets:                0                 0
Octets:                 0                 0
Prot Rejects:           0                 -
```

dn

Lists statistics on the DECnet packets received and transmitted over the PPP interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list dn`

```
DN Statistic          In        Out
------------          --        ---
Packets:              0         0
Octets:               0         0
Prot Rejects:         0         -
```

osicp

Lists statistics for the OSI control protocol. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list osicp`

```
OSICP Statistic          In                  Out
---------------          --                  ---
Packets:                 0                   0
Octets:                  0                   0
Prot Rejects:            0                   -
```

osi

Lists statistics on the OSI packets received and transmitted over the PPP interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list osi`

```
OSI Statistic          In             Out
-------------          --             ---
Packets:               0              0
Octets:                0              0
Prot Rejects:          0              -
```

bvcp

Lists statistics on the Banyan VINES control protocol. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list bvcp`

```
BVCP Statistic           In                Out
--------------           --                ---
Packets:                 0                 0
Octets:                  0                 0
Prot Rejects:            0                 -
```

vines

Lists statistics for the Banyan VINES packets received and transmitted over the PPP interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list vines`

```
Vines Statistic           In                    Out
---------------           --                    ---
Packets:                  10                    13
Octets:                   320                   340
Prot Rejects:             0                     -
```

isrcp

Lists statistics for APPN ISR Control Protocol packets. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list isrcp`

```
APPN ISRCP Statistic      In                    Out
---------------           --                    ---
Packets:                  3                     3
Octets:                   12                    12
Prot Rejects:             0                     -
```

isr

Lists statistics on the APPN ISR packets received and transmitted over the PPP interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list isr`

```
APPN ISR Statistic        In                    Out
-------------             --                    ---
Packets:                  220                   219
Octets:                   1266                  1157
Prot Rejects:             0                     -
```

hprcp

Lists statistics for APPN HPR Control Protocol packets. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list hprcp`

```
APPN HPRCP Statistic      In                    Out
---------------           --                    ---
Packets:                  3                     3
Octets:                   12                    12
Prot Rejects:             0                     -
```

hpr

Lists statistics on the APPN HPR packets received and transmitted over the PPP interface. These fields are the same as those described under the **list ip** command. (See 34-19.)

Example: `list hpr`

```
APPN HPR Statistic         In                    Out
-------------             --                    ---
Packets:                  780                   715
Octets:                   131907                69685
Prot Rejects:             0                     -
```

LLC

Use the **LLC** command to access the LLC monitoring prompt. LLC commands are entered at this new prompt. See “LLC Monitoring Commands” on page 25-1 for an explanation of each of these commands.

Note: This command is available only when APPN is included in the software load.

Syntax: llc

Example: `llc`

```
LLC user monitoring
LLC>
```

Exit

Use the **exit** command to return to the GWCON prompt level.

Syntax: exit

Example: **`exit`**

Point-to-Point Protocol Interfaces and the GWCON Interface Command

The PPP interface traffic is carried by an underlying data-link level device driver. Additional statistics that can be useful when monitoring PPP links may be obtained from the device driver statistics which are displayed using the **interface** command from the GWCON environment. (For more information on the **interface** command, refer to Chapter 6, “The GWCON (Monitoring) Process and Commands” on page 6-1.)

The statistics in this section display when you run the **interface** command from the GWCON environment for the following interfaces used in point-to-point configurations:

Example: **`interface 1`**

```
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
1  1   PPP/0        81620   5D          0         83            0

  Point to Point MAC/data-link on SCC Serial Line interface

  Adapter cable:           V.35 DTE  RISC Microcode Revision:  1


   V.24 circuit: 105 106 107 108 109 125 141
   Nicknames:    RTS CTS DSR DTR DCD RI  LL
   PUB 41450:    CA  CB  CC  CD  CF  CE
   State:        ON  OFF OFF ON  OFF OFF OFF

  Line speed:              unknown
  Last port reset:         1 minute, 54 seconds ago

 Input frame errors:
   CRC error                      0   alignment (byte length)     0
   missed frame                   0   too long (>  2182 bytes)    0
   aborted frame                  0   DMA/FIFO overrun            0
   L & F bits not set             0
Output frame counters:
   DMA/FIFO underrun errors       0   Output aborts sent          0
```

Nt
: Interface number as assigned by software during initial configuration.

Nt'
: Base interface number as assigned by software during initial configuration.

  Note: For dial circuit interfaces, Nt' is different from Nt. For dial circuit interfaces, Nt' indicates the base interface (ISDN or V.25bis) that the dial circuit uses.

Interface No
: Type of interface and its instance number. The Point-to-Point interface type is PPP.

CSR
: Command and status register addresses of the base network.

Vec
: Interrupt vector address.

Self-Test: Passed
: Total number of times the point-to-point interface passed its self-test.

Self-Test: Failed
: Total number of times the point-to-point interface failed its self-test.

Maintenance: Failed
: Total number of maintenance failures.

Adapter cable
: Type of adapter cable that has been configured; for example, V.35 DTE.

V.24 circuit
: Circuits being used on the V.24. Note: The symbol - - - in console output indicates that the value or state is unknown.

Nicknames
: Control signals Note: The symbol - - - in console output indicates that the value or state is unknown.

PUB 41450
: Pin assignments Note: The symbol - - - in console output indicates that the value or state is unknown.

State
: State of the V.24 circuits (on or off). Note: The symbol - - - in console output indicates that the value or state is unknown.

Line speed
: Configured line speed or default value assumed (if line speed is configured as 0).

Last port reset
: Length of time since the port was reset.

CRC error
: The number of packets received that contained checksum errors and as a result were discarded.

Alignment (byte length)
: The number of packets received that were not an even multiple of 8 bits in length and as a result were discarded.

Too long (> 2048 bytes)
: The number of packets that were greater than the configured frame size, and as a result were discarded.

Aborted frame
: The number of packets received that were aborted by the sender or a line error.

DMA/FIFO overrun
: The number of times the serial interface could not send data fast enough to the system packet buffer memory to receive them from the network.

Missed frame
: When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.

L & F bits not set
: On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the `L & F bits not set` counter is incremented, and the buffer is cleared for reuse.

  Note: It is unlikely that the `L & F bits not set` counter will be affected by traffic.

Output Frame Counters:

DMA/FIFO underrun errors
: The number of times the serial interface could not retrieve data fast enough from the system packet buffer memory to transmit them onto the network.

Output aborts sent
: The number of transmissions that were aborted as requested by upper-level software.

Chapter 35. Using and Configuring the Multilink PPP Protocol

The Multilink PPP Protocol (MP) allows you to increase the bandwidth of ISDN B-channels by defining a ***virtual link*** made up of multiple links. The bandwidth of the resulting MP bundle is almost equal to the sum of the bandwidths of the individual links. The advantage is that large data packets transmitted across a single link can now be fragmented, transmitted across multiple links, and rebuilt at the receiving end station. MP helps eliminate bottlenecks in the ISDN portion of your network. MP uses both the Bandwidth Allocation Protocol and the Bandwidth Allocation Control Protocol to, add links to and drop links from, a virtual link.

There are two types of MP links: those that are dedicated and those that are simply enabled. A dedicated MP link is an MP-enabled dial circuit configured as a link to a particular MP interface. If the dial circuit attempts to join another MP bundle, or if MP is not negotiated at all, the software ends the call. An MP-enabled dial circuit that is not dedicated can become a link in any MP bundle. If MP is not negotiated, the dial circuit operates as an independent interface using the dial circuit's configured protocols.

You can configure an Multilink PPP interface that consists of multiple PPP dial circuits as part of the MP bundle. Each of the PPP dial circuit interfaces must use an ISDN base net.

There are also two types of MP interfaces: those that have a dedicated link and those that do not. An MP interface needs a dedicated link in any one of the following situations:

- The link is only for the MP interface
- The MP interface is configured for outbound calls. The dedicated link must then be configured with the destination phone number and caller identification.
- The MP interface is configured to receive a particular inbound call. In this case, the dedicated link is configured with the inbound destination phone number and caller identification.
- The MP interface needs to perform outbound authentication. In this case, all links use the same authentication name.

MP interfaces that do not have a dedicated link must be inbound-only interfaces. These interfaces are similar to the any inbound dial circuit.

The Bandwidth Allocation Protocol (BAP) and its control protocol (BACP) allow an MP interface to increase and decrease its bandwidth by adding and dropping ISDN B-channels. When the bandwidth utilization algorithm determines that a link should be added to the bundle, if there is an available PPP dial-circuit, an available B-channel, and the peer agrees, an additional call is placed.

BAP first searches for any idle dedicated PPP dial circuits for the MP interface, and then for any MP-enabled PPP dial circuit. It will not, however, use a dedicated PPP dial circuit of another MP circuit. The configured maximum number of links on the MP interface will never be exceeded.

Configuring a Multilink PPP Interface

This section shows how to configure a Multilink PPP interface by using an example that configures Multilink PPP with two ISDN dial circuits.

1. Add the two dial circuits and the multilink PPP interface.

```
*t 6

Config>add dev dial-circuit
Adding device as interface 7
Defaulting Data-link protocol to PPP
Use "net 7" command to configure circuit parameters
Config>add dev dial-circuit
Adding device as interface 8
Defaulting Data-link protocol to PPP
Use "net 8" command to configure circuit parameters
Config>add dev multilink-ppp
Adding device as interface 9
Defaulting Data-link protocol to PPP
Use "net 9" command to configure circuit parameters
Config>
```

2. Configure each PPP dial circuit. (See Chapter 49, “Configuring Dial Circuits” on page 49-1.) In this example, the destination, call direction, and LIDs are set for one of the dial circuits.

```
Config>net 7
Circuit configuration
Circuit config:   7>set dest out
Circuit config:   7>set calls outbound
Circuit config:   7>set net 6
Circuit config:   7>
```

3. Enable MP on each dial circuit to be used for MP as follows:

```
Circuit config:   7>encapsulator
Point-to-Point user configuration
PPP 7 Config>enable mp

Enabled as a Multilink PPP Link,
Use as a dedicated Multilink PPP link? [No]: yes
Multilink PPP net for this Multilink PPP link [1]? 9
NOTE: PPP configuration will be obtained from the Multilink PPP
      net.  It is NOT necessary to configure PPP for this net!
```

   Note: You cannot configure PPP parameters for dedicated links from this prompt. Dedicated links use the existing MP interface’s PPP configuration.

   By answering “Yes” to the question “Use as a dedicated Multilink PPP link?” the link becomes dedicated to the specified Multilink PPP interface (9 in this example). In this case, the link ***must*** be used for an MP bundle and ***must*** join the specified MP interface. The link cannot be used as a regular PPP dial circuit.

   Answering “No” to “Use as a dedicated Multilink PPP link?” will allow this PPP dial-circuit to join any MP interface. At least one PPP dial-circuit ***must*** be a dedicated link to an outbound MP interface.

   A dedicated PPP dial circuit obtains all PPP parameters (LCP options, authentication, and others) from its MP interface. MP enabled PPP dial circuits joining the same MP bundle ***must*** negotiate the same LCP parameters and authentication name.

4. Configure the MP interface. The “Dialout MP link net” should be a dedicated PPP dial circuit.

```
Config>net 9
Circuit configuration
MP config:   9>set calls out
Dialout MP link net for this MP Net [0]? 7
MP config:   9>
```

Protocols, BAP, BRS, WAN restoral, WAN reroute, and dial-on-demand are all run on the MP interface and not the PPP dial circuits.

Accessing the MP Configuration Prompt

To access the `MP config >` prompt:

1. Enter **talk 6** at the * prompt.
2. Enter **net *n***, where n is the number of the dial circuit that you enabled to use MP.

   Note: You are now configuring the Multilink PPP interface and not the PPP dial circuit that is part of the MP bundle.

MP Configuration Commands for Multilink PPP Interfaces

Table 35-1 lists the commands available at the `MP config >` prompt.

Table 35-1. MP Configuration Commands

Command	Function
? (Help)	Displays all the MP commands or the options available for a specific command.
Disable	Disables the negotiation of BAP/BACP and bandwidth on demand.
Enable	Enables the negotiation of BAP/BACP and bandwidth on demand.
Encapsulator	Places you in the `PPP config >` prompt so you can change the data-link protocol configuration.
List	Displays the MP interface configuration parameters.
Set	Configures MP interface for inbound or outbound traffic. Also allows you to set the idle timeout and other MP and BAP parameters.
Exit	Returns you to the previous prompt level.

? (Help)

Displays the MP interface commands or lists parameters for specific MP commands.

Syntax: ?

Example: ?

```
DISABLE
ENABLE
ENCAPSULATOR
LIST
SET
EXIT
```

Example: `set ?`

```
CALLS
IDLE
MP
BAP
```

Disable

Use the **disable** command to disable the negotiation of BAP. Disabling BAP prevents the link from allocating additional bandwidth when necessary.

Syntax: disable bap

Example: **`disable bap`**

```
BAP disabled
```

Enable

Use the **enable** command to enable the negotiation of BAP. Enabling BAP allows the link to allocate additional bandwidth when necessary.

Syntax: enable bap

Example: **`enable bap`**

```
BAP enabled
```

Encapsulator

Use the **encapsulator** command to access the PPP link-layer configuration for the Multilink PPP interface.

Syntax: encapsulator

Example: **`encapsulator`**

```
Point-to-Point user configuration
PPP config>
```

List

Use the **list** command to display the current MP configuration.

Syntax: list

Example: **`list`**

```
Idle timer = 0 (fixed circuit)
Outbound calls = allowed
Dialout MP Link net = 7
Max fragment size = 750
Min fragment size = 375
Maximum number of active links = 2
Links associated with this MP bundle:
net number 7
net number 8
BAP enabled
Add bandwidth percentage = 90
Drop bandwidth percentage = 70
Bandwidth test interval (sec) = 15
```

Idle timer

The setting of the idle timer for this circuit in seconds.

A setting of 0 indicates a fixed circuit. A non-zero setting configures a dial-on-demand MP circuit that will be brought down when the circuit is idle for the specified number of seconds The circuit is re-activated when network traffic resumes.

Outbound calls
Specifies whether the interface is configured to initiate outbound calls. If the interface cannot initiate outbound calls, this line is not displayed.

Inbound calls
Specifies whether the interface is configured to initiate inbound calls. If the interface cannot accept inbound calls, this line is not displayed.

Dialout MP link net
The ISDN dial circuit configured to place the first call for an outbound MP circuit.

Max fragment size
Specifies the largest number of bytes of data a packet can contain before the packet is fragmented to be sent over MP links.

Min fragment size
This is the minimum size of the fragments (in bytes) the software creates when a packet exceeds ***Max fragment size***.

Maximum number of active links
Specifies the configured maximum number of links in the MP virtual link (also known as ***bundle***).

Links associated with this MP bundle
Displays the links dedicated to this MP interface.

BAP enabled
Specifies whether BAP is enabled on this interface.

Add bandwidth percentage
The amount of bandwidth utilization at which the software will try to add a new link if BAP is enabled.

Drop bandwidth percentage
The amount of bandwidth utilization at which the software will remove a link from the MP bundle if BAP is enabled.

Bandwidth test interval
The time, in seconds, after which the software will check the bandwidth utilization to determine whether to add or drop a link from the bundle.

Set

Use the **set** command to configure:

- The MP interface for inbound or outbound calls
- The idle timeout
- The MP parameters
- The BAP parameters

Syntax: set bap parameters

calls

idle

mp parameters

bap parameters
Prompts you to specify the BAP add and drop bandwidth percentages and the BAP test interval.

Example: **`set bap parameters`**

```
Add bandwidth % [90]? 80
Drop bandwidth % [70]? 50
Bandwidth test interval (sec) [15]? 25
```

Add bandwidth %

The amount of bandwidth utilization at which the software will try to add a new link.

Valid Values: 1 to 99

Default Value: 90

Drop bandwidth %

The amount of bandwidth utilization at which the software will remove a link from the MP bundle.

Valid values: 1 to 99

Default value: 70

Bandwidth test interval (sec)

The time, in seconds, after which the software will check the bandwidth utilization to determine whether to add or drop a link from the bundle.

Valid Values: 10 to 200 seconds

Default Value: 15

calls

Specifies whether this MP interface will initiate outbound calls, only accept outbound calls, or participate in both types of calls.

Valid values: inbound, outbound, or both

Default value: inbound

Note: If you specify outbound or both, the software will request the net number of the dedicated MP link that will place the first call.

Example: **`set calls outbound`**

```
Dialout MP link net for this MP net []? 4
```

idle Specifies the time period in seconds that an interface can have no protocol traffic at which the MP interface will end calls on all the links.

Valid Values: 0 to 65535

Default Value: 0

Example: **`set idle 60`**

mp parameters

Prompts you to enter the maximum and minimum fragment sizes and the maximum number of active links.

Example: **`set mp parameters`**

```
Max frag size [750]? 675
Min frag size [375]? 300
Max number of active links [2]? 4
```

Max frag size

Specifies the largest of number of bytes of data a packet can contain before the packet is fragmented to be sent over MP links.

Valid Values: 100 to 3000

Default Value: 750

Min frag size
This is the minimum size of the fragments (in bytes) the software creates when a packet exceeds ***Max fragment size***.

Valid Values: 100 to 3000

Default Value: 375

Max number of active links
Specifies the configured maximum number of links in the MP virtual link (also known as ***bundle***).

Valid Values: 1 to 64

Default Value: 2

Exit

Use the **exit** command to return to the previous command prompt.

Syntax: exit

Example: `exit`

Chapter 36. Monitoring Multilink Protocol (MP)

This chapter describes how to monitor specific Multilink PPP interfaces in a device. The chapter includes:

- "Monitoring MP Interface Status"
- "Accessing the MP Monitoring Commands"
- "Multilink PPP Protocol Monitoring Commands"

Monitoring MP Interface Status

To determine the status of all the MP interfaces in your device, use the **configuration** command in ***talk 5*** (see "Configuration" on page 6-6).

Accessing the MP Monitoring Commands

To access the MP monitoring commands:

1. Enter **talk 5** at the * prompt.
2. Enter **net *n***, where ***n*** is the number of the MP interface.

Multilink PPP Protocol Monitoring Commands

Table 36-1 shows the monitoring commands available for an MP interface.

Table 36-1. MP Monitoring Commands

Command	Function
? (Help)	Display the commands available at this command level or the parameters for a specific command.
List	Displays BAP, BACP, and MP statistics, errors, and other information.
Exit	Returns you to the previous command prompt.

? (Help)

Displays the MP interface commands or lists parameters for specific MP commands.

Syntax: ?

Example: ?

```
LIST
EXIT
```

Example: `list ?`

```
BACP
BAP
CONTROL BACP
CONTROL BAP
CONTROL MP
MP
```

List

Use the **list** command to display information about the MP interface including bandwidth allocation statistics.

Syntax: **list** bacp

bap

control bacp

control bap

control mp

mp

Note: The examples that follow assume that the MP interface on this device is net number 6.

bacp

The **list bacp** command lists the statistics for bandwidth allocation control packets which have been sent or received on this MP circuit.

Example:

```
PPP 6> list bacp

BACP Statistic            In          Out
-------------             --          ---
Packets:                  6           8
Octets:                   60          80
Rejects:                  0           -
```

bap The **list bap** command lists the statistics for bandwidth allocation protocol packets which have been sent or received on this MP circuit.

Example:

```
PPP 6> list bap

BAP Statistic               In          Out
-------------               --          ---
Packets:                    3           3
Octets:                     22          37
Call Requests:              1           0
Call Response(ACK):         0           1
Call Resp(NK & FLLNK):      0           0
Call Response(Rej):         0           0
Callback Requests:          0           0
Callback Response(ACK):     0           0
Cllbck Resp(NK & FLLNK):    0           0
Callback Response(Rej):     0           0
Drop Requests:              0           1
Drop Response(ACK):         1           0
Drop Resp(NK & FLLNK):      0           0
Drop Response(Rej):         0           0
Call Status(Success):       1           0
Call Status(Fail):          0           0
```

There are four different responses to a peer's request: ACK, NAK, FULL-NAK, and REJECT.

ACK Indicates the peer's request has been granted.

NAK (NK)
Indicates that the peer's request is supported but not desired at this time. Try again later.

FULL-NAK (FLLNK)

Indicates that the peer s request is supported but because of a resource condition, cannot be granted at this time. The request should not be sent again until the total bandwidth across the MP bundle changes.

REJECT (REJ)

Indicates that the request is not supported.

control bacp

The **list control bacp** command lists the current state of the BACP state-machine within PPP. The state information is identical to that produced for all of the PPP control protocols. Information about favored peer is also listed. Favored peer is used to alleviate BAP packet collisions (when both sides simultaneously initiate requests). During BACP negotiations, each side sends a magic-number and the one with the smallest magic number is the favored peer and should take precendence in the event of a collision. Typically, the call initiator will choose a ***magic number*** of X'1' and the call receiver will choose a magic number of X'FFFFFFFF' establishing the call initiator as the favored peer.

```
PPP 6> list control bacp

BACP State:               Open

BACP Option               Local                    Remote
----------                -----                    ------
Magic Number:             FFFFFFFF                 1
Favorite Peer:            NO                       YES
```

control bap

The **list control bap** command lists the state of the bandwidth allocation protocol and bandwidth on demand. This information includes BAP state, configured bandwidth on demand parameters for adding and subtracting bandwidth, current bandwidth, and information from the last bandwidth poll.

Example:

```
PPP 6> list control bap

BAP State:                                  Ready
Bandwidth test interval (sec):              15
Add bandwidth percentage:                   90
Drop percentage (links-1):                  70
Max # active links in MP bundle:            3
Time since last Bandwidth check (sec):      5
Currently:
    # active links in MP bundle:           1
    Total MP bandwidth (Bytes/sec):         8000
Last Bandwidth Check:
    # active links in MP bundle:           2
    Avg Inbound bandwidth util (%):         12
    Avg Outbound bandwidth util (%):        12
    Drop check: Avg In (%) for links-1:     24
    Drop check: Avg Out (%) for links-1:    24
```

Note: Drop percentage considers current utilization for links - 1

Valid BAP states are:

Closed

BACP is not opened – BAP either is not enabled or not supported by the peer.

Ready
BACP is opened and there is no outstanding request being processed.

Call Req Sent
There is an outstanding call-request that was sent from the local machine.

Callback Req Sent
There is an outstanding callback-request that was sent locally.

Call Placed
As a result of a BAP request to add bandwidth, a call has been placed.

Retry Status Sent
The outgoing call failed to join the MP bundle, a retry status was sent.

No Retry Status Sent
The outgoing call either succeeded or exhausted all retries, a no retry status was sent.

Drop Req Sent
There is an outstanding drop request that was sent locally.

Configured bandwidth-on-demand parameters include add percentage, drop percentage, maximum number of active links in the MP bundle, and the bandwidth polling interval.

A BAP request to add a link to the bundle will be initiated if both the following conditions are met:

- The current number of active links is less than the configured maximum number of links.
- The bandwidth utilization across all links in the MP bundle is greater than the add percentage of the total available bandwidth for the MP bundle.

A BAP request to drop a link from the MP will be initiated if both the following conditions are met:

- The number of active links is greater than one.
- The bandwidth utilization across all links in the MP bundle is less than the drop percentage of the total available bandwidth for the MP bundle for the number of links minus one.

Bandwidth can be polled only when BAP is in the ready state. The information listed from the previous poll will give you an idea of the bandwidth utilization across the MP bundle.

These two sets of information are displayed when a drop can be initiated:

- Bandwidth utilization across the entire bundle
- Bandwidth utilization across number of links minus one

To prevent thrashing, the second set of information is used when determining whether to drop a link.

control mp
The **list control mp** command lists the current state of this MP circuit including the number of active links and bandwidth, the configured maximum

number of links, and statistics for number of dropped packets. Dropped MP packets are classified into four categories:

M
The packet is dropped because a sequence number has not been received and it is less than the minimum sequence number across all links' last received sequence number.

Timeout
The packet is dropped because a sequence number has not been received during a timeout period.

Q depth
The packet is dropped because the maximum queue depth was exceeded.

Seq order
The packet is dropped because the sequence number received was not expected. This occurs when MP receives delayed packet that it has already declared lost.

If a packet is dropped at the network layer, it can be either an M, Timeout, or Q depth packet. These counters are incremented appropriately when a packet is dropped.

```
PPP 6> list control mp

Current # active links in MP bundle:        2
Max # active links in MP bundle:            3
Total MP bandwidth (Bytes/sec):             16000
Dropped Frags (lost - M):                   0
Dropped Frags (timeout):                    0
Dropped Frags (Q depth):                    0
Dropped Frags (seq order):                  0
```

mp The **list mp** command lists the statistics for packets which have been sent or received on this MP circuit. The number of bytes displayed is for pre-decompressed packets if compression was negotiated for the multilink bundle.

```
PPP 6> list mp

MP Statistic               In          Out
------------               --          ---
Bytes (Compressed):        61230       60259
```

Exit

Use the **exit** command to return to the previous command prompt.

Syntax: exit

Example: `exit`

Chapter 37. Using and Configuring a Dial-In Access to LANs (DIALs) Server

A DIALs Server allows remote users to dial in to a LAN and access the resources of the LAN in the same manner as if they were locally attached with a LAN adapter. Similarly, the DIALs Server also allows LAN-attached users to dial out to WAN resources (such as bulletin boards, FAX machines, Internet Service Providers (ISP) and other on-line services) eliminating the need for an analog phone line and modem on their workstation.

The DIALs Server can be configured for both dial-in and dial-out users simultaneously. The IBM DIALs Dial-In Client runs on the remote workstation and provides the dial-in function. Figure 37-1 shows an example of a device used as a DIALs Server supporting the dial-in function.

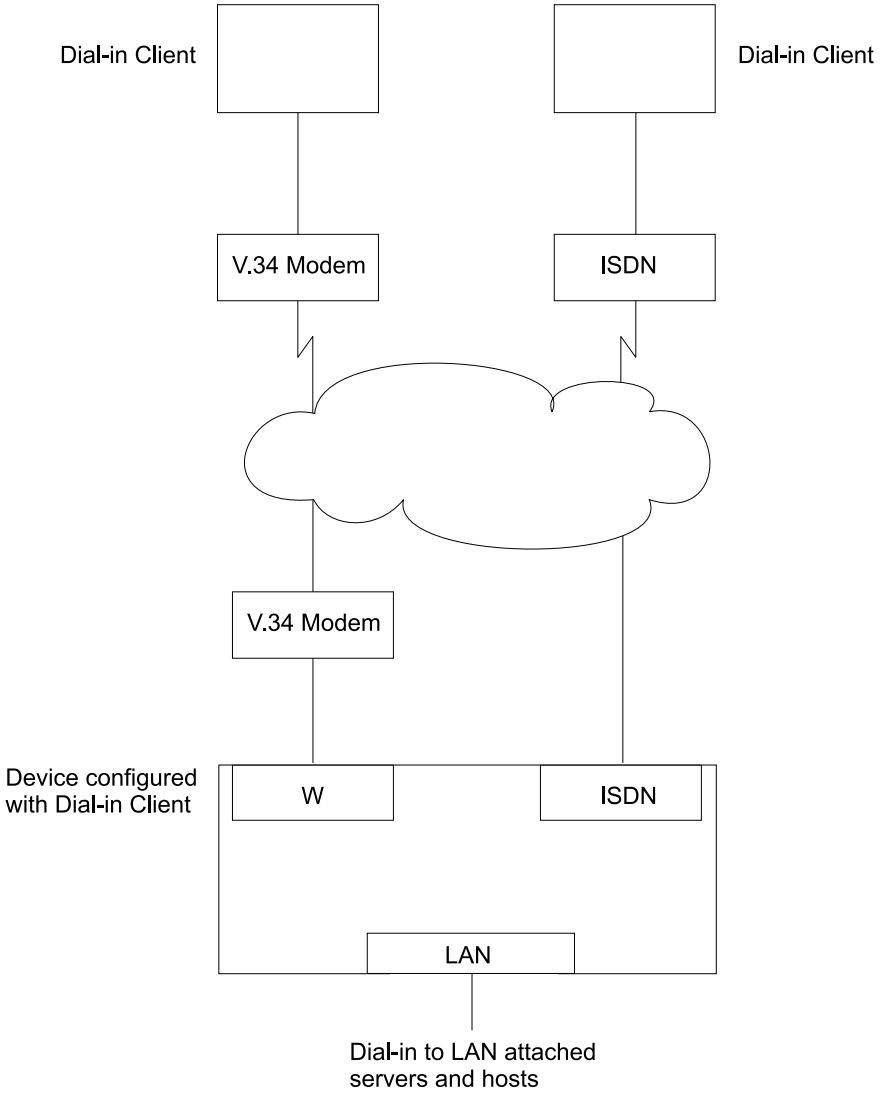


Figure 37-1. An Example of a DIALs Server Supporting Dial-In

The IBM DIALs Dial-Out Client runs on the network-attached workstation and provides the dial-out function. Figure 37-2 on page 37-2 shows an example of a 2210 used as a DIALs Server supporting the dial-out function.

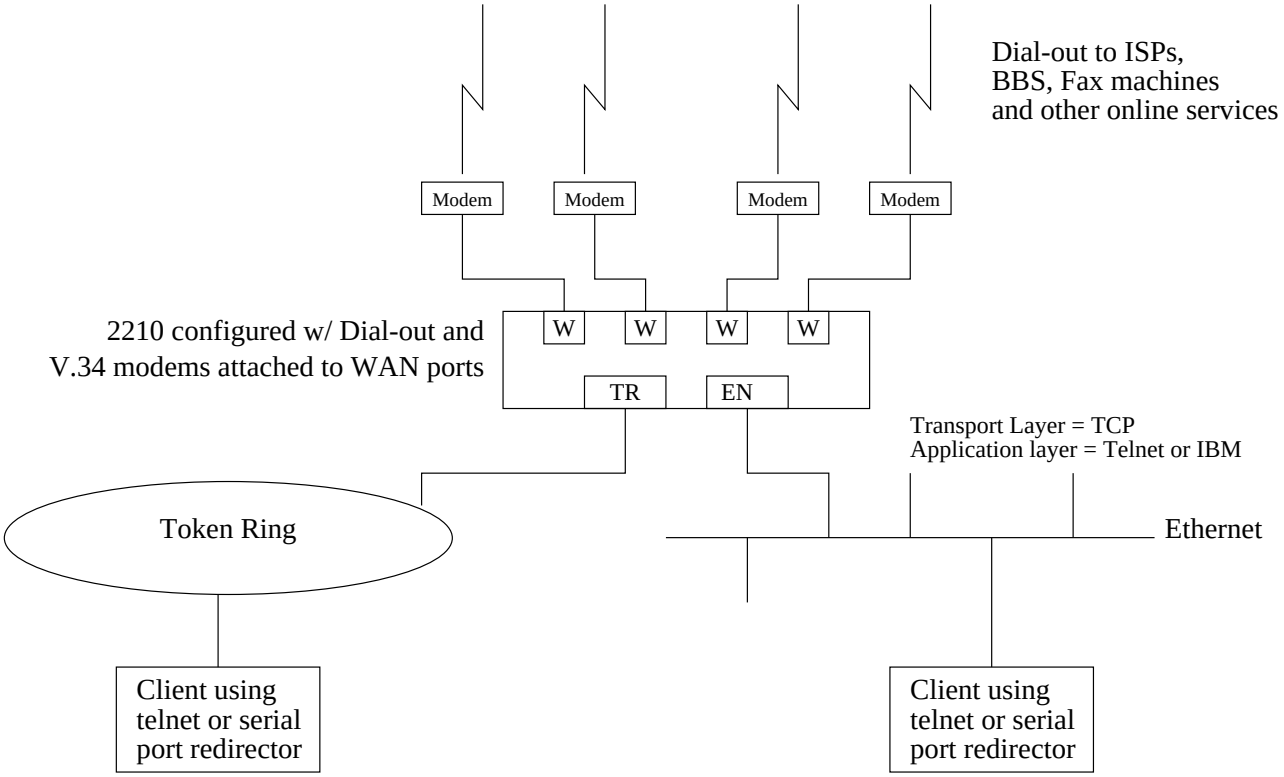


Figure 37-2. An Example of a DIALs Server Supporting Dial-Out

Before Using Dial-In-Access

Before using Dial-In Access, you need:

- A workstation running the IBM DIALs Dial-In Client or another PPP dial-in client (referred to as the ***dial-in client*** or ***PPP dial-in client*** throughout the following sections).
- Completed protocol configurations on the client machine.
- ISDN interfaces or V.34 modems connected to the WAN ports of the 2210 that you wish to use for single user dial-in.
- A fully configured DIALs Server in your LAN.

Configuring Dial-In Access

This section describes how to configure both dial-in and dial-out functions on the DIALs Server. Configuring a client to use dial-in access is described in the documentation associated with the client the workstation uses.

Configuring Dial-In Interfaces

Dial-in interfaces on the 2210 are a special type of dial-circuit. Because most of the settings for a typical dial-circuit are not relevant for single-user dial-in applications, a new device type called ***dial-in*** can be added that sets appropriate defaults for the dial-circuit. Adding a dial-in device also sets up the PPP encapsulator configuration defaults that work with the majority of PPP dial-in clients, including the IBM DIALs Dial-In client. These defaults are described in "Dial Circuit Parameter Defaults for Dial-In Interfaces" and "Dial Circuit PPP Encapsulator Parameters for Dial-In Circuits."

Note: DIALs function can only be enabled on dial-in circuits. Dial-in circuits are only supported when the base net is a V.34 or ISDN net.

Dial Circuit Parameter Defaults for Dial-In Interfaces

Note: Do not override the parameters described in this section. Doing so will prevent the dial-in function from operating correctly. For a complete description of the parameters, see Chapter 49, "Configuring Dial Circuits" on page 49-1.

The following defaults are set when you add a dial-in interface:

- ***Idle time*** is set to 0. Note that a standard circuit is defined as a circuit where the idle timer has no meaning. It will not be a fixed circuit to automatically dial-out. The only time the circuit will dial-out is if a PPP callback has been negotiated or if Multilink PPP has been enabled on this circuit. See "Shiva Password Authentication Protocol (SPAP)" on page 33-9 and Chapter 35, "Using and Configuring the Multilink PPP Protocol" on page 35-1.
- ***Inbound calls*** are allowed. Any inbound is setup because PPP dial-in clients do not use the LID exchange implemented by Nways dial-circuits.
- ***Outbound calls*** are allowed.

  Note: "Outbound" for a dial-in circuit is not the same as a dial-out circuit. See "Before Configuring Dial-Out Interfaces" on page 37-4.

- A default destination address is set up for "default_address" and this address is added to either the list of V.34 address or ISDN addresses. Because these calls are inbound and the only outbound calls will be the result of either a callback or a multilink PPP exchange, the destination address is meaningless. However the address is required for the circuit parameters. Do not delete this address or your circuits will come up disabled.

Dial Circuit PPP Encapsulator Parameters for Dial-In Circuits

Note: For a complete description of the following parameters see Chapter 33, "Using and Configuring Point-to-Point Protocol Interfaces" on page 33-1.

The following defaults are set when you add a dial-in interface:

- Authentication is enabled for SPAP, CHAP, and PPP.
- The PPP MRU is set to 1522. This MRU size is needed for the Windows 3.1, OS/2, and DOS versions of the IBM DIALs Dial-In clients. Do not change this setting unless you know you are not using these clients.
- Automatically enables DIALs on the PPP encapsulator. This turns on some of the features important for Dial-In Access to LANs users such as the NetBIOS Control protocol, Antibes Frame Control protocol, time remaining, SPAP

authentication, callback, LCP identification, and automatic addition and deletion of IP static routes to the client. See Chapter 33, “Using and Configuring Point-to-Point Protocol Interfaces” on page 33-1 for more information on the DIALs features.

Adding a Dial-In Interface

To add a dial-in interface:

1. Configure a V.34 or ISDN base net on one of the available WAN interfaces of the 2210. See Chapter 45, “Using and Configuring the V.34 Network Interface” on page 45-1 and Chapter 47, “Using and Configuring the ISDN Interface” on page 47-1for configuration details.
2. Enter **talk 6** to access the `Config >` prompt.
3. Enter **add device dial-in** at the `Config >` prompt to add the dial-in interface. Adding a device type of dial-in will create a new net with a net number of the current number of nets plus one. You will be asked what base net should be associated with this dial-in net. Enter a configured V.34 or ISDN base net number in response to this question.

Figure 37-3 is an example of defining a dial-in interface.

```
MOS Operator Control

*t 6
Gateway user configuration
Config>li dev
Ifc 0   Ethernet                        CSR  81600, CSR2  80C00, vector 9
        (50 Receive Buffers)
Ifc 1   V.34 Base Net                   CSR  81620, CSR2  80D00, vector 9
        (24 Receive Buffers)
Ifc 2   V.34 Base Net                   CSR  81640, CSR2  80E00, vector 9
        (24 Receive Buffers)
Config>add dev dial-in
Adding device as interface 3
Defaulting Data-link protocol to PPP
Use "net 3" command to configure circuit parameters
Base net for this circuit ;0;? 1
Config>li dev
Ifc 0   Ethernet                        CSR  81600, CSR2  80C00, vector 9
        (50 Receive Buffers)
Ifc 1   V.34 Base Net                   CSR  81620, CSR2  80D00, vector 9
        (24 Receive Buffers)
Ifc 2   V.34 Base Net                   CSR  81640, CSR2  80E00, vector 9
        (24 Receive Buffers)
Ifc 3   PPP Dial-in Circuit
Config>
```

Figure 37-3. Adding a Dial-In Interface

Before Configuring Dial-Out Interfaces

Before configuring and using dial-out interfaces on the 2210, you need:

- IBM Nways software with DIALs support loaded on a 2210.
- A V.34 modem. if connecting to an available WAN port on the 2210. See Chapter 45, “Using and Configuring the V.34 Network Interface” on page 45-1 for configuration information.
- A workstation connected to the LAN that has access to the 2210 DIALs Server.

- Software on the client such as telnet, a telnet redirector or the IBM DIALs Dial-Out clients. IP must be correctly configured on the client in order for the dial-out client to work.

Configuring Dial-Out Interfaces

The following steps describe how to configure a dial-out interface on your device.

1. Connect a V.34 modem to the WAN port that you will use as a dial-out interface.
2. Connect to the console of the 2210 DIALs Server.
3. Enter **talk 6** at the * prompt.
4. Set up a V.34 interface. See Chapter 45, "Using and Configuring the V.34 Network Interface" on page 45-1 for details.
5. Add a dial-out interface using the **add device dial-out** command. When prompted for the interface, use an available V.34 interface number.

   Notes:

   a. Multiple circuits can be configured on top of a v34 base net. However, only one circuit can be active at any given time.

   b. The software defines a V34 address called ***default_address***. Do not delete this address as it is required by dial-out and dial-out will not work without it.
6. Configure the PPP authentication server, if you are using the IBM DIALs Dial-Out client, and add PPP users as described in "PPP Authentication Protocols" on page 33-7. The added PPP users should have dial-out enabled. Dialing out using telnet does not require authentication, therefore do not configure authentication for telnet sessions.
7. Configure the global dial-out parameters. Enter **feature dial** (see "Feature" on page 3-25) to enter the Dial-In-Access configuration environment.

   In this environment you can configure the dial-out inactivity timer, the dial-out server name, modem pools, and other parameters.
8. Restart the device.

Configuring Modem Pools

Modem pools are defined as a group of modems which appear to the user as one modem. When the user needs to dial-out, the first available modem in this pool is used. Modem pools are created in the 2210 DIALs Server by defining groups of dial-out interfaces with the same portname. By default, all dial-out interfaces are named "ALL_PORTS" which creates a modem pool. Naming the dial-out interfaces individually enables a user to select a particular modem to dial-out.

To configure a modem pool:

1. Enter **talk 6** at the * prompt.
2. Enter **net *n***, where ***n*** is the number of the dial-out interface as defined in step 4. This action places you in the configuration environment for the interface.
3. Enter **encapsulator** (see "Encapsulator" on page 49-3) at the `Circuit Config>` prompt. This action places you in the dial-out configuration environment.

4. Enter **set portname** at the `Dial-out Config>` prompt. This action will prompt you for the name of the port (up to 30 characters). If you specify an existing port name, the modem is added to the pool with that name.
5. Restart the 2210.

DIALs Configuration

This section contains commands used to configure a DIALs Server. Other related commands appear in:

- "Add" on page 3-12
- "Feature" on page 3-25
- "Set" on page 3-32
- "Entering and Exiting the ELS Configuration Environment" on page 8-2

Dynamic Host Configuration Protocol (DHCP)

The Dynamic Host Configuration Protocol (DHCP) was developed to provide configuration parameters to hosts on a network. Among other configuration parameters, DHCP has a mechanism for allocation of network addresses to hosts.

The Proxy DHCP feature acts as a client *on behalf* of a dial-in PPP user. This allows the device to obtain an IP address lease for the duration of the dial-in session, or until the lease expires. The IP address that is allocated from the DHCP server is communicated to the dial-in client through PPP IPCP (see "IP Control Protocol" on page 33-13 for a description of IPCP). The dial-in client software has no knowledge that DHCP was used to allocate an IP address, and thus requires no DHCP activation of any kind.

Proxy DHCP requires that at least one DHCP server be configured and accessible from the router.

Proxy DHCP requires that the addresses being allocated to dial-in users be within the same subnet of a directly connected LAN. In a typical configuration, this requires enabling proxy ARP subnet routing to allow the router to answer ARP requests to hosts on the local network on behalf of the dial-in clients.

Basic DHCP Setup

The most basic configuration calls for a single DHCP server on the same network as the router, with dial-in addresses to be leased within the same subnet as this LAN.

When the client dials in, a lease for an IP address is obtained from the DHCP server and used in IPCP negotiation with the client.

1. Connect 2210 and DHCP to the same LAN.
2. Configure and start the DHCP server (see your DHCP server's documentation for how to setup your server to lease IP addresses. Remember, the IP addresses to be leased MUST be within a subnet of a directly connected LAN and proxy ARP must be enabled on the 2210).
3. The typical setup for Proxy DHCP disables Client-Specified, Userid and Interface IP Address Negotiation options:

```
Dials Config>list ip
DIALs client IP address specification:
Client : disabled
UserID : disabled
Interface : disabled
DHCP Proxy : enabled
```

This simply states that the user must get his or her IP address from the DHCP server. The router should disallow the user from specifying his or her own address as well as ignoring any IP address that may or may not be configured in the Userid or Interface section.

4. Add DHCP server (`Dials Config>` **add dhcp 10.0.0.111**)
5. Set dial-in client software to *Server assigned*.

   Notes:

   a. *Server assigned* configuration varies among different dial-in client implementations.

   b. The client software should not be configured to obtain its address from DHCP. The client should obtain its address by sending an address of 0.0.0.0 to IPCP on the initial configure request.
6. For this setup, let the DHCP GATEWAY ADDRESS default to 0.0.0.0.

Multiple Hops to DHCP Server

The configured DHCP server(s) should be IP addresses which are reachable from the connected router. You should always be able to ping the server from the remote access box.

When the DHCP server is located multiple hops away, the server needs to know an address to reply to, and to indicate which pool to allocate an IP address from. The pool to allocate an IP from is important because the DHCP server could be utilized to serve addresses to a number of subnets and there must be some indication as to which pool of addresses to select from. The DHCP Gateway Address (*giaddr*) is used for this (the terminology is based on the definition given in RFC 2131). The *giaddr* must be an address that is local to the 2210, such as the token ring or Ethernet LAN port. Also, since the *giaddr* is the address which the DHCP server will use to reply, make sure you can ping this address from the DHCP server itself.

Multiple DHCP Servers Network

You can configure multiple DHCP servers for redundancy. When you configure multiple servers, the Proxy DHCP client asks all servers for an address and accepts the first response received. If any of the DHCP servers are more than one hop away, or are connected to a subnet which is not associated with the addresses in its pool, then *giaddr* must be configured. See “Multiple Hops to DHCP Server.”

While there can be more than one DHCP server offering addresses, it is important to not allow the pool of addresses configured at each server to overlap. Further, because there is only one *giaddr* for the DHCP server to respond to and perform a lookup with, each pool of address must be in the same subnet as each other.

Dynamic Domain Name Server (DDNS)

A Domain Name Server (DNS) maps IP addresses to hostnames and is typically static in nature. Dynamic DNS is a feature that, when used with a DDNS DHCP server and a DNS server, enables DHCP to dynamically update the DNS server with an IP address and hostname mapping. This feature may only be used in conjunction with Proxy DHCP.

When you enable Dynamic DNS on the 2210 and you configure a hostname in the user profile (see "PPP Authentication Protocols" on page 33-7), this hostname is passed as option 81 (DDNS) to the DHCP SERVER. If you configured the DHCP server correctly for DDNS, the DHCP server updates the DDNS server with the IP address that it leased to the router and the hostname that the router sent to it. This allows other users to access the dial-in client through the hostname rather than requiring the client to know the dynamically chosen IP address.

DIALs Global Configuration Commands

Enter **feature dials** at the `Config>` prompt to access the DIALs (`DIALs Config>`) global parameter configuration environment. Table 37-1 lists the available commands.

Table 37-1. DIALs Global Configuration Commands

Command	Function
? (Help)	Use the ? (help) command to list the commands that are available from the current prompt level. You can also use ? to list the options of a particular command.
Add	Adds a (Dynamic Host Configuration Protocol) DHCP server to the list of DHCP servers.
Delete	Deletes a DHCP server from the list.
Disable	Disables IP address negotiation, dial-out protocols, SPAP Banner, and Dynamic DNS.
Enable	Enables type of IP address negotiation, dial-out protocols, SPAP Banner, and Dynamic DNS.
List	Lists the dial-out configuration including server name, inactivity timeout, and the protocols enabled for dial-out.
Set	Sets time-allowed, dhcp gateway address, NetBIOS Name Server addresses, Dynamic Name Server addresses and dial-out inactivity timer, and dial-out server-name.
Exit	Returns you to the previous prompt level.

? (Help)

Use the **? (Help)** command to display the commands available in the current command available or to display the parameters for a specific command.

Syntax: ?

Example: ?

```
Add
Delete
Disable
Enable
List
Set
Exit
```

Example: set ?

```
Dial-out
DNS
NBNS
DHCP-Gateway-Interface
Time-allowed
```

Add

Use the **add** command to add a new Proxy DHCP server to a list of servers. The list contains the IP addresses of the DHCP servers that will, in turn, lease IP addresses to the dial-in clients. Multiple servers may be added for redundancy. The maximum number of servers is 20.

Syntax: add dhcp-server *ipaddress*

Example:

```
DIALs Config> add dhcp-server
DIALs Proxy DHCP server address [0.0.0.0]? 10.0.0.1
```

Delete

Use the **delete** command to delete an existing Proxy DHCP server from the list of servers.

Syntax: delete dhcp-server *ip address*

Example:

```
DIALs Config> delete dhcp-server
Enter the address to be deleted [0.0.0.0]? 10.0.0.1
```

Disable

Use the **disable** command to disable IP address negotiation, dial-out protocols, SPAP Banner, and Dynamic DNS.

Syntax: disable

dial out . . .

dynamic-dns

ip-address-negotiation . . .

spap-banner

dial-out *type*

Disables the use of dial-out with either telnet or IBM DIALs Dial-Out clients. You can specify:

dials Disables all IBM DIALs Dial-Out clients

telnet Disables all telnet clients.

To disable both types of clients you must enter the disable dial-out command for each type. Disabling both types of clients disables dial-out on the 2210.

IP-address-negotiation *type*

Disables various IPCP address negotiation techniques. You can specify any of the following:

- Client-specified – The router will not allow the client to specify an IP address to be used by the router for that PPP connection. The client must request, and able to accept, an address offered by the router. In turn, one of the other three methods, “userid,” “interface,” or “DHCP-Proxy” must be enabled for the router to obtain an address to offer the client.

- Userid – The router will not offer an IP address based on the authenticated user's profile (see "Configuring PPP Authentication" on page 33-9 for more information).
- Interface – The router will not offer an address defined by the PPP interface. See "IP Control Protocol" on page 33-13 for more information.
- DHCP-proxy – The router will not query a DHCP server for an address defined by the PPP interface. See "Dynamic Host Configuration Protocol (DHCP)" on page 37-6 for more information.

See "IP Control Protocol" on page 33-13 for a description of these techniques.

dynamic-dns
Disables the sending of DHCP option 81 for the user's hostname. See "Dynamic Domain Name Server (DDNS)" on page 37-8 for more information.

spap-banner
Disables the sending of a SPAP banner to a remote user authenticated with SPAP. See "Shiva Password Authentication Protocol (SPAP)" on page 33-9 for more information.

Enable

Use the **enable** command to enable IP address negotiation, dial-out protocols, SPAP Banner, and Dynamic DNS.

Syntax: en**able**

dial-out . . .

dynamic-dns

ip-address-negotiation . . .

spap-banner

dial-out *type*
Enables the use of dial-out with either telnet or IBM DIALs Dial-Out clients. By default, both types of clients are enabled. You can specify:

dials Enables all IBM DIALs Dial-Out clients

telnet Enables all telnet clients.

IP-address-negotiation *type*
Enables various IPCP address negotiation techniques. You can specify any of the following:

- Client-specified – Allows the client to specify the address it wants to use. This takes precedence over "userid," "interface," and "dhcp-proxy."
- Userid – The router will look in the authenticated user profile for an IP address. If the address is non-zero, it will be offered to the client. This takes precedence over "interface" and "dhcp-proxy."
- Interface – The router will look at the IPCP settings for the interface. If the address configured is non-zero, it will be offered to the client. This takes precedence over "dhcp-proxy."
- DHCP-proxy – The router will query a DHCP server for an IP address lease. If unable to obtain a lease, IPCP will fail.

See “IP Control Protocol” on page 33-13 for a description of these techniques.

dynamic-dns
Disables sending of DHCP option 81 for the user’s hostname. See “Dynamic Domain Name Server (DDNS)” on page 37-8 for more information.

spap-banner
Enables the sending of a SPAP banner to a remote user authenticated with SPAP. The command will prompt for the contents of the banner. See “Shiva Password Authentication Protocol (SPAP)” on page 33-9 for more information.

List

Use the **list** command to display the current configuration. The DHCP state and lease times can be monitored for each net from the Point-to-Point console. See 34-7 for an example.

Syntax: list all

dhcp-servers

dial out

dynamic-dns

ip-address-negotiation

name-servers

spap-banner

time-allowed

Example: list all

```
DIALs config>li all
DIALs client IP address specification:
Client     :  enabled
UserID     :  enabled
Interface  :  enabled
DHCP Proxy :  disabled


Note: Proxy DHCP is currently disabled
Configured DHCP servers:           1.1.1.1          2.2.2.2
DHCP Gateway (giaddr): 0

Dynamic DNS: Disabled

Primary Domain Name Server    (DNS) : none
Primary NetBIOS Name Server  (NBNS) : none
Secondary Domain Name Server  (DNS) : none
Secondary NetBIOS Name Server (DNS) : none

Time allowed for connections: unlimited

SPAP BANNER is :Welcome to my world.

Box-level dial-out settings
Inactive timer:                              15
Transport Protocols enabled for dial-out:  TELNET DIALs
Server name:                                 2210_DIALS_SERVER
```

The example shows the following:

DIALs client IP address specification
Displays the IP address negotiation techniques and whether they are enabled. You would receive this section of the display and the section containing the box-level dial-out settings in response to the **list ip-address-negotiation** command.

Configured DHCP servers
Displays the list of IP addresses currently configured as DHCP servers. This section also lists the interface being used for the DHCP gateway. You would receive this section of the display in response to the **list dhcp-servers** command.

Dynamic Name Servers
Displays whether Dynamic DNS is enabled. You would receive this section of the display in response to the **list dynamic-dns** command.

primary domain server (dns)
This line and the following lines display the configured primary and secondary name servers. You would receive this section of the display in response to the **list name-servers** command.

time allowed
Displays the maximum amount of time (in minutes) for dial-out users. You would receive this section of the display in response to the **list time-allowed** command.

spap banner
Displays the contents of the spap banner. You would receive this section of the display in response to the **list spap-banner** command.

Set

Use the **set** command to set the time-allowed, dhcp gateway address, NetBIOS Name Server addresses, Dynamic Name Server addresses and dial-out inactivity timer, and dial-out server-name

Syntax: **set** dhcp-gateway-address

dial-out . . .

dns . . .

nbns . . .

time-allowed

dhcp-gateway-address ***interface# ipaddress***
Sets the IP address associated with the DHCP gateway. DHCP uses the address as:

1. An address to which DHCP replies
2. An indication of the pool of addresses from which DHCP allocates an IP address

If the DHCP server is not on a directly attached LAN interface, then you must configure this address the address of one of the LAN interface that is directly connected to the DHCP server. See “Dynamic Host Configuration Protocol (DHCP)” on page 37-6 and the definition of “giaddr” in RFC 1541 for more information.

dial-out *parameter*

Sets the inactivity timer or server name for dial-out nets. ***Parameter*** can be:

inactivity-timer

Sets the dial-out inactivity timer for dial-out nets. This is defined as the amount of time, in minutes, that a user can be connected without data traffic over the connection. For example, if the inactivity-timer is set to 5 minutes and during any 5 minute interval, no data is received or transmitted, the connection will be dropped and the modem will become available. The default is 0, which means that the inactivity timer is disabled and the connection will be maintained indefinitely.

servername

Sets the name of the dial-out server. This can be any string up to 30 characters in length. The default is “2210_DIALS_SERVER.” This is the name that the IBM DIALs Dial-Out clients see when they use the “Chooser” application to discover dial-out servers. This parameter has no meaning for telnet dial-out clients.

dns *type ipaddress*

Configures the primary and secondary domain name servers (DNS). ***Type*** can be:

primary

Sets the IP address of the primary DNS server for the dial-in client to use. This value is negotiated during IPCP for some dial-up clients (particularly Windows 95).

secondary

Sets the IP address of the secondary DNS server for the dial-in client to use. This value is negotiated during IPCP for some dial-up clients (particularly Windows 95).

nbns *type ipaddress*

Configures the primary and secondary NetBIOS name servers. ***Type*** can be:

primary

Sets the IP address of the primary NetBIOS name server.

secondary

Sets the IP address of the secondary NetBIOS name server.

time-allowed

Sets the time allowed for PPP dial-in user and dial-out users. This parameter defines the maximum amount of time (in minutes) that a user can be connected. The default value is 0, which means the user can be connected for an unlimited amount of time.

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **exit**

Dial-Out Interface Configuration Commands

To access the dial-out interface parameter environment:

1. Enter **talk 6** at the * prompt.
2. Enter **net *n*** at the Config > prompt.
3. Enter **encapsulator** at the Circuit config: n> prompt.

Table 37-2 lists the commands available from the dial-out config> prompt.

Table 37-2. Dial-Out Interface Configuration Commands

Command	Function
? (Help)	Displays the commands available at the current prompt level or the parameters for a specific command.
Set	Defines the port name associated with a modem.
Exit	Exits the dial-out configuration process and returns you to the previous prompt level.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
Set
Exit
```

set ?

```
portname
```

Set

Use the **set** command to define the port name for a modem.

Syntax: set portname *name*

portname

Defines the name of the port associated with a modem. Use this name to define ***modem pools.*** The name can be up to 30 characters in length.

Default value: ALL_PORTS

Example: dial-out config>**set portname localcalls**

Exit

Use the **exit** command to return to the Circuit config: **n**> prompt.

Syntax: exit

Example: **exit**

Chapter 38. Monitoring Dial-In-Access Interfaces

This chapter describes how to monitor dial-in-access interfaces using the DIALs monitoring commands. The chapter includes:

- "Monitoring Dial-In Interfaces"
- "Monitoring Dial-Out Interfaces"

Monitoring dial-in-access interfaces is similar to monitoring other interfaces. To access the interface monitoring prompt:

1. Enter **talk 5** at the * prompt.
2. Enter **net *n***, where ***n*** is the network number of the dial-out interface.

Monitoring Dial-In Interfaces

Monitoring dial-in interfaces is the same as monitoring other PPP dial circuits. For details, see Chapter 34, "Monitoring Point-to-Point Protocol Interfaces" on page 34-1.

Monitoring Dial-Out Interfaces

Table 38-1 lists the commands available when monitoring dial-out interfaces.

Table 38-1. Dial-Out Interface Monitoring Commands

Command	Function
? (Help)	Displays the commands available at this prompt level.
Clear	Resets the statistics for this dial-out interface.
List	Lists the current state of the dial-out interface, the number of bytes transmitted and received on this interface, and the client's current parameters.
Exit	Returns you to the previous prompt level.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
CLEAR
LIST
EXIT
```

Clear

Use the **clear** command to reset the statistics for the number of octets received and transmitted by this interface.

Syntax: clear
Example: **clear**

```
Statistics reset.
```

List

Use the **list** command to display current state of the dial-out interface. The **list** command always displays the current state of the dial-out net, the time since the state change, and the number of bytes received and transmitted.

Syntax: list

Example for inactive interface:

```
list

Dial-out Settings for current session:

Dial-out state is DOWN
Time since change           = 52 minutes and 34 seconds

Dial-out Octets transmitted = 0
Dial-out Octets received    = 0

Session down, no valid settings
```

Note: When a client connects to a dial-out port using telnet, no user name is present because the server did not perform any authentication.

Example for active interface:

```
list

Dial-out Settings for current session:

Dial-out state is UP
Time since change           = 3 seconds

Dial-out Octets transmitted = 14
Dial-out Octets received    = 765

Current user                = not available
Time allowed for user       = unlimited
Inactivity timer for port   = 10 minutes
Line speed                  = 57600
Current DTR state           = DTR ON
Current dial-out protocol   = TELNET
Options negotiated:
        Will Suppress Go Ahead
        Wont' Echo characters
```

Example for an active IBM DIALs Dial-Out client:

```
list

Dial-out Settings for current session:

Dial-out state is UP
Time since change           = 12 seconds

Dial-out Octets transmitted = 11
Dial-out Octets received    = 756

Current user                = ebooth
Time allowed for user       = unlimited
Inactivity timer for port   = 10 minutes
Line speed                  = 57600
Current DTR state           = DTR ON
Current dial-out protocol   = DIALs
```

Exit

Use the **exit** command to return to the GWCON > prompt.

Syntax: exit

Example: **exit**

Chapter 39. Configuring SDLC Relay

This chapter describes the Synchronous Data Link Control (SDLC) Relay configuration commands. The chapter includes the following sections:

For further information on when to use DLSw SDLC versus SDLC Relay, refer to "Relationship to the SDLC Relay Function" in the "Using and Configuring DLSw" chapter of *Protocol Configuration and Monitoring Reference Volume 1 for Nways Multiprotocol Routing Services Version 2.1*

Accessing the SDLC Relay Configuration Environment

To access the SDLC relay (SRLY) configuration environment:

1. At the `Config>` prompt, enter **set data-link srly**.
2. Enter the interface number.
3. To configure the SRLY interface, enter the **network** *interface#* command. The `SRLY` *`interface#`* `Config>` prompt is displayed when **network** ***interface#*** is entered:

```
Config>network 2
SDLC relay interface user configuration
SRLY 1 Config>
```

4. To configure the SRLY protocol parameters, enter the **protocol sdlc** command. The `SDLC Relay config>` prompt is displayed when **protocol sdlc** is entered:

```
Config>protocol sdlc
SDLC Relay protocol user configuration
SDLC Relay config>
```

Basic Configuration Procedure

This section outlines the minimum configuration steps required to get the SDLC Relay protocol up and running. For further configuration information and explanation, refer to the configuration commands described in this chapter.

Note: You must restart the router for new configuration changes to take effect.

- *Adding a number.* You must add a number to a group of primary or secondary ports using the **add group** command. The default number for this command is 1.
- *Adding a local port.* This identifies the interface that you are using for the local port. This also assures that no IP address is configured for the interface that you select. Use the **add local-port** command.
- *Adding a remote port.* This identifies the port directly connected to the remote side of the serial line. Use the **add remote-port** command.

SDLC Relay Configuration Commands

This section summarizes and then explains the SDLC Relay configuration commands. Both the **network** and **protocol** parameters for SDLC relay are documented in this chapter.

The SDLC Relay configuration commands allow you to specify router parameters for interfaces transmitting SDLC Relay frames. Restart the router to activate the configuration commands. Table 39-1 shows the commands for both the **network sdlc** and **protocol sdlc**.

Table 39-1. SDLC Relay Configuration Commands Summary

Command	Network SRLY	Protocol SDLC	Function
? (Help)	yes	yes	Lists all of the SDLC Relay configuration commands or lists the options associated with specific commands.
Add		yes	Adds groups, local ports, and remote ports.
Delete		yes	Deletes groups, local ports, and remote ports.
Disable		yes	Disables groups and ports.
Enable		yes	Enables groups and ports.
List	yes	yes	Displays entire SDLC Relay and group specific configurations.
Set	yes		Sets the link parameters and remote station parameters.
Exit	yes	yes	Exits the SDLC Relay configuration environment and returns to the CONFIG environment.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
Add
Delete
Disable
Enable
List
Set
Exit
```

Add

Use the **add** command to add group numbers, local ports, and remote ports.

Syntax: add group
local-port
remote-port

group
Assigns a number to a group of primary or secondary ports added to the router.

Example: `add group`

```
Group number: [1]? 1
```

Group number The group number that you are designating for the port.

local–port
Identifies the interface that you are using for the local port.

Example: `add local–port`

```
Group number: [1]? 1
Interface number: [0]? 2
(P)rimary or (S)econdary:[S]? p
```

Group number The group number for the port. This number must match one of the **add group** parameters configured previously.

Interface number The interface number of the router that designates the local port.

Primary or Secondary Designates the port type, primary (P) or secondary (S).

remote–port
Identifies the IP address of the port directly connected to the serial line on the remote router.

Example: `add remote–port`

```
Group number: [1]? 1
IP address of remote router:[0.0.0.0]? 128.185.121.97
(P)rimary or (S)econdary:[S]? s
```

Group number The group number for the port. This number must match one of the add group parameters configured previously.

IP address of remote router
Identifies the IP address of the interface on the remote router.

Primary or Secondary Designates the port type, primary (P) or secondary (S).

Delete

Use the **delete** command to remove group numbers, local ports, and remote ports.

Syntax: delete group . . .
local-port . . .
remote–port

group *group#*
Removes a group (group#) of SDLC Relay configured ports.

Example: `delete group 1`

local–port *interface#*
Removes the local port for the specified interface (interface#).

Example: `delete local–port 2`

remote–port

Removes the remote port for the specified group.

Example: `delete remote–port`

```
Group number: [1]? 1
(P)rimary or (S)econdary:[S]? S
```

Group number	The group number for the remote port.
Primary or Secondary	Designates the port type, primary (P) or secondary (S).

Disable

Use the **disable** command to suppress relaying for an entire relay group or a specific relay port.

Syntax: disable group . . .
port

group *group#*

Suppresses transfer of SDLC Relay frames to or from a specific group (group#).

Example: `disable group 1`

port

Suppresses transfer of SDLC Relay frames to or from a specific local port.

Example: `disable port`

```
Group number: [1]? 2
(P)rimary or (S)econdary:[S]? s
```

Group number	The group number of the port that you want to disable.
Primary or Secondary	Designates the port type, primary (P) or secondary (S).

Enable

Use the **enable** command to turn on data transfer for an entire group or a specific local interface port.

Syntax: enable group . . .
port

group *group#*

Allows transfer of SDLC Relay frames to or from the specified group (group#).

Example: `enable group 1`

port

Allows transfer of SDLC Relay frames to or from the specified local port.

Example: `enable port`

```
Group number: [1]? 2
(P)rimary or (S)econdary:[S]? s
```

Group number	The group number of the port that you want to enable.
Primary or Secondary	Designates the port type, primary (P) or secondary (S).

List (for network SRLY)

Use the **list** command to display the configuration of a specific group or of all groups.

Syntax: list

Example: `list`

```
Maximum frame size in bytes = 2048
Encoding:  NRZ
Idle State:  Flag
Clocking:  External
Cable Type:  RS-232 DTE
Speed (bps):  0
Transmit Delay Counter:  0
```

Maximum frame size in bytes
: Maximum frame size that can be sent over the link. The maximum frame size must be large enough to accommodate the largest frame and the 15 byte SRLY header.

Encoding
: The transmission encoding scheme for the serial interface. Scheme is NRZ (non-return to zero) or NRZI (non-return to zero inverted).

Idle State
: The data link idle state: flag or mark.

Clocking
: The type of clocking: internal, external.

Cable Type
: The serial interface cable type.

Speed (bps)
: Lists the speed of the transmit and receive clocks.

Transmit Delay Counter
: Number of flags sent between consecutive frames.

List (for protocol SDLC)

Use the **list** command to display the configuration of a specific group or of all groups.

Syntax: list all
group . . .

all
: Displays the configurations of all local ports.

Example: `list all`

```
                SDLC Relay Configuration

Group Number     Port Status      Net   SDLC Station    IP Address
                                 Number address (hex)
-------------- ----------------- ------ -------------  ------------
    1 (E)      Local   PRMRY (D)   2
    1 (E)      Remote SCNDRY (E)                      128.185.452.11
    2 (D)      Local   PRMRY (D)   1
    2 (D)      Remote SCNDRY (D)                      128.185.450.31
```

Group Number
: Indicates the group number and the status of the group, enabled (E) or disabled (D).

Port Status	Indicates the type of port (local/remote primary/secondary) and its status, enabled (E) or disabled (D).
Net Number	Indicates the device number of the local port. This number matches the number displayed using the Config list devices command.
IP Address	Indicates the IP address of the remote port.

group *group#*

Displays the configuration of a specified group.

Example: `list group 1`

```
             SDLC Relay Configuration

 Group Number     Port Status      Net   SDLC Station    IP Address
                                  Number address (hex)
-------------- -----------------  ------ -------------  ------------
    1 (E)      Local   PRMRY (D)   2
    1 (E)      Remote SCNDRY (E)                     128.185.452.11
```

Group Number	Indicates the group number and the status of the group, enabled (E) or disabled (D).
Port Status	Indicates the type of port (local/remote primary/secondary) and its status, enabled (E) or disabled (D).
Net Number	Indicates the device number of the local port. This number matches the number displayed using the Config list devices command.
IP Address	Indicates the IP address of the remote port.

Set

Use the **set** command to configure the SRLY parameters.

Syntax: set
- cable
- clocking
- encoding
- frame-size
- idle
- speed
- transmit-delay

cable

Sets the cable used on the serial interface. The options are:

- RS-232 DTE
- RS-232 DCE
- V35 DCE
- V35 DTE
- V36 DTE
- X21 DCE
- X21 DTE

A DTE cable is used when you are attaching the router to some type of DCE device (for example, a modem or a DSU/CSU).

A DCE cable is used when the router is acting as the DCE and providing the clocking for direct attachment.

clocking *internal* or *external*

Configures the SRLY link's clocking. To connect to a modem or DSU, configure clocking as external. To connect directly to another DTE device, use a DCE cable, set the clocking to internal, and configure the clock speed. For internal clocking, you must enter a valid line speed in the range 2400 - 2048000 bits per second.

Example: `set clocking internal`

encoding *nrz* or *nrzi*

Configures the SRLY interface's encoding scheme as NRZ (Non-Return to Zero) or NRZI (Non-Return to Zero Inverted). NRZ is the default.

Example: `set encoding nrz`

frame-size

Configures the maximum size of the frames that can be transmitted and received on the data link. If this value is set to a larger value than that specified with the add remote-secondary command, then this value is changed to reflect that maximum. The IBM 2210 generates an ELS message warning the user that this value is changing. The user will continue receiving this ELS message until it is changed in the SRAM configuration. Valid entries are shown in Table 39-2.

Note: The frame size must be large enough to accommodate the largest frame received plus a 15–byte SRLY header.

Table 39-2. Valid Values for Frame Size in Set Frame-Size Command

Minimum	Maximum	Default
128	18000	2048

Example: `set frame-size 4096`

idle flag

Configures the transmit idle state for framing on the SRLY interface. The default is the flag option which provides continuous flags (7E hex) between frames.

Example: `set idle flag`

The link will receive a flag idle transparently.

idle mark

Configures the transmit idle state for framing on the SRLY interface. The mark option puts the line in a marking state (OFF, 1) between frames.

Example: `set idle mark`

The link will receive a mark idle transparently.

speed

For internal clocking, this command specifies the speed of the transmit and receive clock lines. The range of speeds supported is 2400 - 2048000 bits per second. to determine the link speeds you can set for the

transmit-delay *value*

Allows the insertion of a delay between transmitted packets. This command ensures a minimum delay between frames so that it is compatible with older, slower serial devices at the other end. This value is specified as the number of

flag bytes that should be sent between consecutive frames. The range is 0 - 15. The default is 0.

Example: `set transmit-delay 6`

Exit

Use the **exit** command to exit the SDLC Relay configuration process and return to the CONFIG environment.

Syntax: exit

Example: `exit`

Chapter 40. Monitoring SDLC Relay

This chapter describes how to use the Synchronous Data Link Control (SDLC) Relay console commands. The chapter includes the following sections:

Accessing the SDLC Relay Console Environment

To monitor information related to the SDLC Relay interface, access the interface console process by doing the following:

1. Enter the **status** command to find the PID for GWCON. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the **talk** command and the PID for GWCON. For example:

   ```
   * talk 5
   +
   ```

   The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter GWCON, press **Return** again.
3. At the GWCON prompt, enter the **configuration** command to see the protocols and networks for which the router is configured. For example:

   ```
   + configuration
   ```

   See page 6-6 for more sample output from the **configuration** command.
4. Enter the **protocol sdlc** command. For example:

   ```
   + prot sdlc
   SDLC Relay>
   ```

   The SDLC Relay prompt is displayed on the console. You can then view information about the SDLC Relay ports by entering the SDLC Relay console commands.

SDLC Relay Console Commands

This section summarizes and then explains the SDLC Relay console commands. The SDLC Relay console commands allow you to view parameters for interfaces transmitting SDLC Relay frames. The `SDLC Relay>` prompt is displayed for all SDLC Relay console commands. Table 40-1 on page 40-2 shows the commands.

Table 40-1. SDLC Relay Console Commands Summary	
Command	Function
? (Help)	Lists all the SDLC Relay console commands or lists the options associated with specific commands.
Clear-Port-Statistics	Clears SDLC Relay statistics for the specified port.
Disable	Temporarily suppresses groups and ports.
Enable	Temporarily turns on groups and ports.
List	Displays entire SDLC Relay and group specific configurations.
Exit	Exits the SDLC Relay console process and returns to the GWCON environment.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
Clear-port-statistics
Disable
Enable
List
Exit
```

Example: **list ?**

```
all
group
```

Clear-Port-Statistics

Use the **clear-port-statistics** command to discard the SDLC Relay statistics for all ports. The statistics include counters for packets forwarded and packets discarded.

Syntax: clear-port-statistics

clear-port-statistics
Clears port statistics gathered since the last time you restarted the router or cleared statistics.

Example: **clear-port-statistics**

```
Clear all port statistics? (Yes or No): Y
```

Disable

Use the **disable** command to suppress data transfer for an entire group or a specific relay port. SRAM (static read access memory) does not permanently store the effects of the **disable** console command. Therefore when you restart the router, the effects of this command are erased.

Syntax: disable group . . .
port

group *group#*
Suppresses transfer of SDLC Relay frames to or from a specific group (group#).

Example: **`disable group 1`**

port *interface# primary-or-secondary*
Suppresses transfer of SDLC Relay frames to or from a specific local port.

Example: **`disable port`**

```
Interface number: [0]? 2
(P)rimary or (S)econdary: [s]? P
```

Interface number Indicates the interface number of the local port that you want to disable.

Primary or Secondary Indicates whether the port is a primary or secondary.

Enable

Use the **enable** command to turn on data transfer for an entire group or a specific local interface port. SRAM does not permanently store the effects of the **enable** console command. Therefore when you restart the router, the effects of this command are erased.

Syntax: enable group . . .
port

group *group#*
Allows transfer of SDLC Relay frames to or from the specified group (group#).

Example: **`enable group 1`**

port
Allows transfer of SDLC Relay frames to or from the specified local port.

Example: **`enable port`**

```
Interface number: [0]? 2
(P)rimary or (S)econdary: [s]? P
```

Interface number Indicates the interface number of the local port that you want to enable.

Primary or Secondary Indicates whether the port is a primary or secondary.

List

Use the **list** command to display the configuration of a specific group or of all groups.

Syntax: list all
group . . .

all
Displays the configurations of all local ports.

Example: **`list all`**

```
                    SDLC Relay Configuration

Group  Port Status       Net    Packets    IP Address
Num                      Num   fwrd  disc

1 (E)  Local PRMRY (E)    2    2880   57
1 (E)  Remote SCNDRY (E)       4860   13   128.185.452.11
2 (D)  Local PRMRY (D)    1    0      0
2 (D)  Remote PRMRY (D)        0      0    128.185.450.31
```

Group Number Indicates the group number and the status of the group, enabled (E) or disabled (D).

Port Status Indicates the type of port (local/remote primary/secondary) and its status, enabled (E) or disabled (D).

Net Number Indicates the device number of the local port. This number matches the number displayed using the `Config>` **list devices** command.

Packets (fwrd and disc) Indicates how many packets were forwarded (fwrd) and discarded (disc) for that port.

IP Address Indicates the IP address of the remote port.

group *group#*

Displays the configurations of a specified group.

Example: **`list group 1`**

```
                SDLC Relay Configuration

Group  Port Status        Net  Packets    IP Address
Num                       Num  fwrd disc

1 (E)  Local PRMRY (D)    2    2880   57
1 (E)  Remote SCNDRY (E)       4860   13  128.185.452.11
```

Exit

Use the **exit** command to exit the SDLC Relay console process and return to the GWCON environment.

Syntax: exit

Example: **`exit`**

SDLC Relay Interfaces and the GWCON Interface Command

While SDLC Relay interfaces have their own console processes for monitoring purposes, the router also displays complete statistics for installed network interfaces when you use the **interface** command from the GWCON environment. (For more information on the **interface** command, refer Chapter 6, The GWCON (Monitoring) Process and Commands.)

Chapter 41. Configuring SDLC Interfaces

This chapter describes the SDLC configuration commands and includes the following sections:

You enter SDLC configuration commands at the `SDLC # Config>` prompt, where # identifies the interface you specify with the network command. Changes made to the routers configuration do not take effect immediately, but become part of the router's static configuration memory when it is restarted.

Accessing the SDLC Configuration Environment

Use the CONFIG process to change the configuration of the router. The new configuration takes effect when the router is restarted.

To enter the configuration process:

1. Enter **talk 6** (or **t 6**), at the OPCON (*) prompt. This brings you to the `CONFIG>` prompt as shown in the following example:

   ```
   MOS Operator Control

   * talk 6

   CONFIG>
   ```

   If the `CONFIG>` prompt does not appear immediately, press the **Enter** key again. All SDLC configuration commands are entered at the `SDLC config>` prompt.

2. At the `Config>` prompt, enter the **set data-link sdlc** command. When prompted, enter the name of the interface to associate with the SDLC device.

   ```
   Config>set data-link sdlc
   Interface number [0]? 2
   Config>
   ```

3. Next, enter the **network** command, plus the number of an SDLC interface that you entered earlier.

   ```
   Config>network 2
   SDLC 2 Config>
   ```

Refer to Chapter 1, "Getting Started (Introduction to the User Interface)" on page 1-1 for information related to the configuration environment.

Basic Configuration Procedure

This section outlines the minimum configuration required for SDLC to be usable by DLSw or by APPN.

Before beginning any configuration procedure, use the **list device** command from the config process to list the interface numbers of different devices. At the config prompt, select the interface you want to configure by entering either: **network** *interface number* or **n** *interface number*. If you need any further configuration

command explanations, refer to the configuration commands described in this chapter.

SDLC Configuration Requirements

In addition to the SDLC-specific configuration procedures and commands described in this chapter, you need to configure SDLC in the DLSw or APPN protocol. Only one protocol at a time, DLSw or APPN, may run over a given SDLC interface. In other words, link stations on a given SDLC interface cannot be divided between APPN and DLSw. If a DLSw configuration and an APPN configuration exist for the same SDLC interface, the first protocol to come active will own the SDLC interface.

SDLC Configuration Commands

The SDLC configuration commands allow you to create or modify the SDLC interface configuration. This section summarizes and describes the commands you can issue from the `SDLC Config>` prompt within the network configuration console. Defaults for any command and its parameters are displayed on the console, they are enclosed in brackets immediately following the prompt.

Note: In addition to configuring SDLC using the commands described in this chapter, you also need to configure SDLC in the DLSw or APPN protocol.

2210 supports SDLC connections over RS-232, X.21, and V.35 serial interfaces. Table 41-1 lists SDLC configuration commands and their function.

Table 41-1. SDLC Configuration Commands Summary

Command	Function
? (Help)	Lists the configuration commands or lists any parameters associated with that command.
Add	Adds an SDLC end station.
Delete	Removes an SDLC end station.
Disable	Prevents connections to one of the SDLC link stations.
Enable	Allows connections to one of the SDLC link stations.
List	Displays configured information for one of the SDLC link stations.
Set	Configures specific interface and link-station information.
Exit	Exits the `SDLC config>` process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

```
Set
Add
Disable
Delete
Enable
List
Exit
```

Add

Use the **add** command to add an end station. The router is, by default the primary end station. If you do not use this command and if you configured an SDLC station in DLSw or in APPN, the end station is added for you. The software assigns the following defaults to the station:

- Maximum BTU is maximum allowable by the interface
- Tx and Rx Windows are 7 for MOD 8, 127 for MOD 128

If the defaults are satisfactory, you do not need to add SDLC station.

Syntax: add station

Example: **add station**

```
Enter station address (in hex) [C3]?
Enter station name [SDLC_C3]?
Include station in group poll list ([Yes] or No):
Enter max packet size [2009]?
Enter receive window [7]?
Enter transmit window [7]?
```

Enter station address	The station's SDLC address in the range 01 - FE.
Enter station name	The name designation of the SDLC station (maximum characters is 8).
Include station in group poll list	Select whether or not to include this station in the group poll list for this link. The SDLC software supports the IBM 3174 group poll function for SDLC secondary station. You must add a group poll address using the **set link group-poll** command for this parameter to have an affect.
Enter max packet size	The maximum packet size that can be sent to or received from the remote link station. This value cannot be greater than that specified for the link. This value is configured with the **set link frame-size** command.
Enter receive window	The maximum number of packets that the router can receive without sending a response.
Enter transmit window	The maximum number of packets that the router can transmit without receiving a response.

Delete

Use the **delete** command to remove the specified end station (station name or address) from the SDLC configuration. The router is considered the primary end station (default).

Syntax: delete station *name* or *address*

Example: **delete station c1**

Disable

Use the **disable** command to prevent connections from being created with a SDLC link station.

Syntax: disable link
station . . .

link
Prevents the transmitting and receiving of data to all configured SDLC link stations on the interface.

Example: `disable link`

station *name* or *address*
Prevents the transmitting and receiving of data to the specified end station (station name or address).

Example: `disable station c1`

Enable

Use the **enable** command to enable connections to remote SDLC link stations.

Syntax: enable link
station

link
Allows subsystems in the router (for example, DLSw) to use SDLC's facilities.

Example: `enable link`

station *name* or *address*
Allows connections to the specified secondary remote end station (link station name).

Example: `enable station c1`

List

Use the **list** command to display configuration information on one or all SDLC link stations.

Syntax: list link
station *name* or all

link
Displays the SDLC interface's configuration.

Example: `list link`

```
Link configuration for: LINK_2   (ENABLED)

Role:          SECONDARY        Type:         POINT-TO-POINT
Duplex:        FULL             Modulo:       8
Idle state:    FLAG             Encoding:     NRZ
Clocking:      EXTERNAL         Frame Size:   2048
Speed:         0                Group Poll:   F3
Cable          V.36 DTE

Timers:     XID/TEST response:  2.0 sec
            SNRM response:      2.0 sec
            Poll response:      0.5 sec
```

```
           Inter-poll delay:   0.2 sec
           RTS hold delay:     DISABLED
           Inter-frame delay:  DISABLED
           Inactivity timeout  30.0 sec

 Counters:  XID/TEST retry:  8
            SNRM retry:      6
            Poll retry:      10
```

Link configuration	The name and status of SDLC link station that are in the router's configuration.
Role	The primary, secondary, or negotiable role for link stations that you configure using the **set link role** command.
Type	The type of link, MULTIPOINT or POINT-TO-POINT.
Duplex	Duplex configuration, HALF or FULL.
Modulo	The sequence number range to use on the link: MOD 8 (0-7) or MOD 128 (0 - 127).
Idle state	The bit pattern (FLAG or MARK) transmitted on the line when the interface is not transmitting data.
Speed	The physical data rate of the interface. When the clocking is internal, this is the data rate generated by the internal clock.
Group Poll	Address used for the group poll feature for multipoint link configurations. Secondary stations having group inclusion coded as yes will respond to unnumbered polls received from this address. This address must be non-null for the group poll feature to be in effect for any secondary stations under this link. Each secondary station will still have a unique station address in addition to the group address.
Cable	Specifies the type of cable in use (RS-232, V.35, V.36, or X.21).
Encoding	Configures the SDLC transmission encoding scheme as NRZ (Non-Return to Zero) or NRZI (Non-Return to Zero Inverted).
Clocking	Interface clocking, EXTERNAL or INTERNAL.
Frame Size	The maximum frame size that can be sent over the interface.
Timers:	All the timers listed below have a 100ms resolution.
XID/TEST resp.	The time to wait for an XID or TEST response message before retransmitting the XID or TEST frame. A value of 0 indicates that the router will continue to retry indefinitely.
SNRM response	The maximum time to wait for an UA response message before the station retransmits SNRM(E).
Poll response	The maximum time to wait for a response from any polled station before retrying.
Inter-poll delay	The amount of time the router (configured with a primary role) waits after receiving a response, before polling the next station.
RTS hold delay	The amount of time that the primary router waits before dropping RTS low after the transmission of a frame. The RTS hold delay parameter is specific to half-duplex operation.
Interframe delay	The number of flags sent between frames.
Inactivity timeout	For idle NRM/E secondary stations, sets the time after which the interface changes the station to its recovery state. A 0 (zero) causes the station to remain idle indefinitely.

Counters:

XID/TEST retry	The maximum number of times the router sends an XID or TEST frame without receiving a response before timing out. A value of 0 indicates that the router will retry indefinitely.
SNRM	The maximum number of times the router will send an SNRM(E) frame without receiving a response before timing out. A value of 0 indicates that the router will retry indefinitely.
Poll retry	The maximum number of times the router polls the station without receiving a response before timing out. A value of 0 indicates that the router will continue to retry indefinitely.

station *all* or *address* or *link station name*

Displays information for the specified SDLC link station or for all link stations.

Example: `list station c1`

```
Address   Name     Status   Max BTU   Rx Window   Tx Window
-------  ------    ------   -------   ---------   ---------
C1(00)  SDLC_C1    Enabled     2005         7           7
```

Example: `list station all`

```
Address   Name      Status     Max BTU  Rx Window  Tx Window
-------  --------  ----------  -------  ---------  ---------
C1(00)   SDLC_C1     ENABLED      2005       7          7
C3(F3)   SDLC_C3     DISABLED     2009       7          7
```

Address	The address of the SDLC link station. The address in parentheses is the group address of the station. A (00) indicates that a group address is not defined.
Name	The character string name designation of SDLC link station.
Status	The status of the SDLC link station, ENABLED or DISABLED.
Max BTU	The frame size limit of the station. This frame size must not be larger than the maximum Basic Transmission Unit (BTU) packet size configured with the **set link frame-size** command.
Rx Window	The size of the receive window.
Tx Window	The size of the transmit window.

Set

Use the **set** command to configure specific information for one or all SDLC link stations.

Syntax: set
- link cable
- link clocking
- link duplex . . .
- link encoding . . .
- link frame-size
- link group poll ...
- link idle . . .
- link inactivity ...
- link inter-frame delay
- link modulo . . .
- link name
- link poll . . .

link role . . .
link rts-hold
link snrm
link speed
link type . . .
link xid/test
station *address* . . .

link cable *type*
Sets the cable connected to this interface. The options are V.36 and the following DCE and DTE types: RS-232, V.35, and X.21.

A DTE cable is used when you are attaching the router to some type of DCE device (for example, a modem or a DSU/CSU).

A DCE cable is used when the router is acting as the DCE and providing the clocking for direct attachment.

Example: `set link cable V35 dte`

link clocking *internal* or *external*
Configures the SDLC link's clocking. To connect to a modem or DSU, set clocking external. To connect directly to another DTE device, use a DCE cable, set the clocking to internal, and configure the clock speed. For internal clocking, you must set the line speed in the range 2400 to 2048000 bits per second.

Example: `set link clocking internal`

link duplex *full* or *half*
Configures the SDLC line for full-duplex or half-duplex.

Example: `set link duplex full`

link encoding *nrz* or *nrzi*
Configures the SDLC transmission encoding scheme as NRZ (Non-Return to Zero) or NRZI (Non-Return to Zero Inverted). NRZ is the default.

Example: `set link encoding nrz`

link frame-size
Configures the maximum size of the frames that can be transmitted and received on the data link. Valid entries are shown in Table 41-2.

Table 41-2. Valid Values for Frame Size in Link Frame-Size Command

Minimum	Maximum	Default
128	18000	2048

Set the link frame size greater than the maximum packet size that you configured with the **set station xxx max packet** command. Otherwise, the router automatically resets the maximum packet size to the link frame size and issues the following ELS message:

```
SDLC.054: nt 3 SDLC/0 Stn xx-MaxBTU too large for Link adjusted (4096->2048)
```

Example: set link frame-size

link group-poll
Sets a group poll address for secondary stations on the link. The SDLC software supports the IBM 3174 group poll function. Use the **add station** or

the **set station group-inclusion** command to include a station in the group poll list.

Example: `set link group-poll`

```
Enter group poll address (in hex) [00:]?f3
Group poll support enabled
```

link idle flag
Configures the transmit idle state for SDLC framing. The default is the flag option which provides continuous flags (7E) between frames.

Example: `set link idle flag`

The link will receive a flag idle transparently.

link idle mark
Configures the transmit idle state for SDLC framing. The mark option puts the line in a marking state (OFF, 1) between frames.

Example: `set link idle mark`

link inactivity *#-of-seconds*
For idle NRM/E secondary stations, sets the time after which the interface changes the station to its recovery state. The range is 0 to 7200 seconds. The default is 30. A 0 (zero) causes the station to remain idle indefinitely.

Example: `set link inactivity`

```
Enter secondary link station inactivity timeout :[30.0]?
```

link inter-frame delay
Allows the insertion of a delay between transmitted packets. This command ensures a minimum delay between frames so that it is compatible with older, slower serial devices at the other end. The delay is specified in terms of the number of flags that should be sent between consecutive frames. The range is 0 to 15 flags and 0 (in other words, no flags) is the default value.

Example: `set link inter-frame delay`

```
Transmit Delay Counter [0]?
```

link modulo 8 or 128
Specifies the sequence number range to use on the link: MOD 8 (0-7) or MOD 128 (0 - 127). Default is 8.

Note: When you change this value, the window sizes become invalid. Use the **set station** command to change the receive window and transmit window sizes. Valid window sizes for mod 8 are 0 through 7; for mod 128 they are 8 through 127.

Also, at connection start-up, an SNRME rather than a SNRM is used and supervisory frame headers are expanded by an additional byte.

Example: `set link modulo 8`

link name
Establishes a character string for the link that you are configuring. This parameter is for informational purposes only.

Example: `set link name`

```
Enter link name: [LINK_0]?
```

link poll delay
Configures the time delay between each poll that is sent over the interface.

Example: `set link poll delay`

```
Enter delay between polls [0.2]?
```

link poll retry
Configures the number of times the interface retries to poll the secondary SDLC link station before it closes the connection.

Example: `set link poll retry`

```
Enter poll retry count (0 = forever) [10]?
```

link poll timeout
Configures the amount of time the interface waits for a poll response before timing out.

Example: `set link poll timeout`

```
Enter poll timeout [2.0]?
```

link role *primary* or *secondary* or *negotiable*
Configures the interface as an SDLC primary, secondary, or negotiable link station (default is primary).

Notes:

1. For DLSw, ***negotiable*** uses X'FF' (broadcast address) for the initial poll.

   When using broadcast address to negotiate the role, the link uses a default SDLC configuration.

   When ***primary*** is the link role, the link performs an initial poll to a specific address.

2. For APPN point-to-point or negotiable, the broadcast address is used for the initial poll. For primary multipoint, the specific address is used.

Example: `set link role primary`

link rts-hold
The time to hold Request-to-Send (RTS) high after transmitting a frame. This setting is for half-duplex mode. This setting has no effect in full-duplex mode.

Example: `set link rts-hold`

```
Enter RTS hold duration after transmit complete [0.0]?
```

link snrm *timeout* or *retry*
Configures the following SNRM(E) information for primary stations:

timeout	The time to wait for an Unnumbered Acknowledgements (UA) response before retransmitting an SNRM(E).
retry	The number of times to retransmit an SNRM(E) without receiving a response before giving up.

Example: `set link snrm timeout`

```
Enter SNRM response timeout [2.0]?
```

Example: `set link snrm retry`

```
Enter SNRM retry count (0=forever) [6]?
```

link speed
For internal clocking, this command specifies the speed of the transmit and receive clock lines.

Example: `set link speed`

```
Line Speed [64000]?
```

link type *multipoint* or *point-to-point*
Configures the SDLC link to either a multipoint link or a point-to-point link.

Example: `set link type multipoint`

link xid/test *timeout* or *retry*
Configures the following XID/test information for primary stations:

timeout	The maximum amount of time to wait for an XID or TEST frame response before retransmitting the XID or TEST frame.
retry	The maximum number of times an XID or TEST frame is resent before giving up. A 0 (zero) causes the router to retry indefinitely.

Example: `set link xid/test timeout 10`

remote-secondary *address* or *link_station_name address* <*argument*>
Changes the remote station's SDLC address in the range 02 - FE.

Example: `set remote-secondary SDLC_C1 address ce`

station *address* or *name* address
Changes the station's SDLC address in the range 01 to FE.

Example: `set station c1 address`

```
Enter station address (in hex) [C1]?
```

station *address* or *link station name* group-inclusion *no* or *yes*
For SDLC secondary stations, set whether to include this station in the group poll list for this link. For this to be effective, add a group poll address using the **set link group-poll** command.

Example: `set station c1 group-inclusion yes`

station *address* or *name* max-packet
The maximum size of the packet that the station can receive (default: 2048). Do not set the maximum packet size larger than the link frame size that is configured with the **set link frame-size** command; if you do, the router automatically resets the maximum packet size to the link frame size and issues the following ELS message:

```
SDLC.054: nt 3 SDLC/0 Stn xx-MaxBTU too large for Link adjusted (4096->2048)
```

Example: `set station c1 max-packet`

```
Enter max packet size [2048]?
```

station *address* or *name* name
The name of the SDLC station.

Example: `set station c1 name`

```
Enter station name [SDLC_C1]?
```

station *address* or *name* receive window
The maximum number of frames the router can receive before sending a response. The range is 1 to 7. The default is 7.

Example: `set station c1 receive-window`

```
Enter receive window [7]?
```

station *address* or *name* transmit-window

The maximum number of frames the router can transmit before receiving a response frame. The range is 1 to 7. The default is 7.

Example: `set station c1 transmit-window`

```
Enter transmit window [7]?
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: **`exit`**

Chapter 42. Monitoring SDLC Interfaces

This chapter describes the SDLC console commands. Some of these commands are identical to those in the SDLC Config> process, and they allow you to dynamically configure the SDLC interface without permanently affecting the SRAM configuration.

This chapter includes the following sections:

Changes made at the configuration command console (SDLC CONFIG>) become part of the SRAM configuration when you restart the router.

Conversely, SDLC monitoring commands entered within the SDLC monitoring process take effect immediately. However, changes made with monitoring commands do not become part of the router's static configuration. When the router is restarted, the effects of the monitoring commands are overwritten by the router's static configuration. Monitoring consists of these actions:

- Monitoring the protocols and network interfaces that are currently in use by the router
- Making real-time changes to the SDLC configuration without permanently affecting the SRAM configuration
- Displaying ELS (Event Logging System) messages relating to router activities and performance

Accessing the SDLC Monitoring Environment

The monitoring environment is the GWCON process. To enter the GWCON process:

1. Enter **talk 5** (or **t 5**) at the OPCON (*) prompt. This brings you to the GWCON (+) prompt as shown in the following example:

```
MOS Operator Control

* talk 5
+
```

2. Next, enter the **network #** command using the number that identifies the interface that you previously configured for the SDLC device.

```
+ network 2
SDLC Console
SDLC-2>
```

You enter all GWCON (Monitoring) commands at the + prompt.

Refer to Chapter 1, "Getting Started (Introduction to the User Interface)" on page 1-1 for information related to the monitoring environment.

SDLC Console Commands

This section summarizes and then explains the SDLC console and related commands. Use these commands to gather information from the database. Table 42-1 lists SDLC console commands and their function.

Table 42-1. SDLC Console Commands Summary

Command	Function
? (Help)	Lists all the SDLC monitoring commands or any options associated with those commands.
Add	Adds an SDLC link station
Clear	Clears the counters on the SDLC interface.
Delete	Dynamically removes an SDLC link station.
Disable	Disables connections to one SDLC link station.
Enable	Enables connections to one SDLC link station.
List	Displays SDLC link stations configurations and link station information.
Set	Configures specific interface and link station information.
Test	Tests the link between the router and the SDLC link station.
Exit	Exits the SDLC console process.

? (Help)

Use the **?** (help) command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **?**

```
ADD
CLEAR Counters
DELETE
DISABLE
ENABLE
LIST
SET
TEST
EXIT
```

Add

Use the **add** command to add an end station. The router is, by default the primary end station. If you do not use this command and if you configured an SDLC station in DLSw or APPN, the end station is added for you.

Syntax: add station

For an example and for additional information on the **add** command, see “Add” on page 41-3.

Clear

Use the **clear** command to clear counters for the interface, for a station, or for all stations. Use the **list all stations** command to list stations.

Syntax: clear link
station ...

link *name* or *address*
Clears the counters for an SDLC interface.

Example: `clear link`

link *name* or *address*
Clears the counters for an SDLC interface.

Example: `clear link c1`

station *name* or *address* or all
Clears counters for a specific station or for all stations.

Example: `clear station c1`

Delete

Use the **delete** command to terminate an existing SDLC connection without affecting the SDLC configuration in SRAM. This command terminates any SDLC session that may be in progress on the link station. The router is considered the primary end station by default.

Syntax: delete station *name* or *address*

Example: `delete station c1`

Disable

Use the **disable** command to disable connection establishment on one or all SDLC link stations without affecting the SDLC configuration in SRAM. The **disable** command also terminates any existing connection to the station.

Syntax: disable link
station . . .

link
Prevents connection on all configured SDLC link stations on the interface by terminating all connections.

Example: `disable link`

station *name* or *address*
Prevents connection to the specified end station (link station name) by terminating any existing connection.

Example: `disable station C1`

Enable

Use the **enable** command to enable connection establishment with remote SDLC link stations without affecting the SDLC configuration SRAM.

Syntax: enable link
station . . .

link
Allows subsystems (for example, DLSw) to use SDLC's facilities.

Example: `enable link`

station *name* or *address*
Allows connections to the specified end station.

Example: `enable station c1`

List

Use the **list** command to display statistics specific to the data link layer and the interface.

Syntax: list link configuration
link counters
station . . .

link configuration
Displays information for all configured SDLC link stations on the interface.

Example: `list link configuration`

For an example and for additional information on the **list** command, see “List” on page 41-4.

link counters
Displays information for the SDLC counters since the last router restart or the last clear counters.

Example: `list link counters`

```
     I-Frames     I-Bytes     Re-Xmit     UI-Frames     UI-Bytes
     --------     -------     -------     ---------     --------
Send       0           0           0             0           0
Recv       0           0                         0           0

          RR         RNR         REJ           UP
     -------     ------      ----       ---------
Send       0          0           0             0
Recv       0          0           0             0
```

I-Frames Total number of Information frames received and transmitted.

I-Bytes Total number of Information bytes received and transmitted.

Re-Xmit Total number of frames that were retransmitted.

UI-Frames Total number of Unnumbered Information frames received and transmitted.

UI-Bytes Total number of Unnumbered Information bytes received and transmitted.

RR Total number Receive-Ready (RRs) received and transmitted.

RNR Total number Receive-Not-Ready (RNRs) received and transmitted.

REJ Total number of Rejects received and transmitted.

UP Unnumbered Polls (group poll) received and transmitted.

station *all* or *address* or *link station name*

Displays the status of the specified SDLC link station or all stations. The software displays an * next to the stations that were not explicitly configured

using the **add station** command but were added to the configuration because they were defined and activated in the protocol layer (DLSw or APPN).

Displays information for the specified SDLC link station (link station name) on the interface.

Example: `list station all`

```
Address    Name      Status    Max BTU   Rx Window     Tx Window
--------   -----     ------    -------   ---------     ---------
 C1(00)    SDLC_C1   IDLE        2048       7             7
 C2(F3)    SDLC_C2   ENABLED     2048       7             7
```

Example: `list station c1`

```
Address   Name      Status     Max BTU     Rx Window   Tx Window
-------  ------    --------    -------     ---------   ---------
* C1(00) SDLC_C1    ENABLED       2048         7           7
```

Address The address of the SDLC link station. The address in parentheses is the group address of the station. A (00) indicates that a group address is not defined.

Name The character string name designation of SDLC link station.

Status The status of the SDLC link station:

Enabled	Enabled, but not allocated
Idle	Allocated, but not in use
Connected	Connected
Disconnected	Disconnected
Connecting	Connection establishment in progress.
Discnectng	Disconnection in progress
Recovering	Attempting to recover from a temporary data link error.

Max BTU The frame size limit of the remote station. This frame size must not be larger than the maximum Basic Transmission Unit (BTU) packet size configured with the **set link frame-size** command. The default is 2048 bytes.

Rx Window The size of the receive window.

Tx Window The size of the transmit window.

station *name* or *address* counters

Displays frame transmit and receive counts for the specified link station.

Example: `list station c1 counters`

```
         I-Frames    I-Bytes     Re-Xmit   UI-Frames    UI-Bytes   XID-Frames
        ----------  ----------  ---------  ----------  ----------  ----------
 Send            9         384          0           0           0           6
 Recv           29       42792                      0           0           3

           RR          RNR         REJ         TEST        SNRM        DISC
        ----------  ----------  ---------  ----------  ----------  ----------
 Send          598           0          0           0           1           0
 Recv          587           0          0           0           0           0

           UA          DM         FRMR       UP
        ----------  ----------  ---------  ---------
 Send            0           0          0          0
 Recv            1           0          0          0
```

I-Frames Number of information frames received and transmitted

I-Bytes	Number of information bytes received and transmitted
Re-Xmit	Number of frames retransmitted
UI-Frames	Number of Unnumbered Information frames received and transmitted
UI-Bytes	Number of Unnumbered Information bytes received and transmitted
XID-Frames	Number of Exchange Identification frames received and transmitted
RR	Number of Receive Ready frames received and transmitted
RNR	Number of Receive Not Ready frames received and transmitted
REJ	Number of Rejects received and transmitted
TEST	Number of Test frames received and transmitted
SNRM	Number of Set Normal Response Mode frames received and transmitted
DISC	Number of Disconnect frames received and transmitted
UA	Number of Unnumbered Acknowledgment frames received and transmitted
DM	Number of Disconnected Mode frames received and transmitted
FRMR	Number of Frame Reject frames received and transmitted
UP	Unnumbered Polls (group poll) received and transmitted.

Set

Use the **set** command to dynamically configure specific information for one or all SDLC link stations without affecting the SRAM configuration. In the SDLC monitoring environment, the **set** command can be executed only on disabled links or stations. All time values are entered in seconds, with a 0.1 second resolution.

Syntax: set
- link modulo . . .
- link name
- link poll . . .
- link role . . .
- link rts-hold
- link snrm(e)
- link type . . .
- link xid/test
- station . . .

link modulo

Dynamically changes the range of sequence numbers to be used on the data link without affecting the SRAM configuration. Modulo 8 specifies a sequence number range 0 - 7, and modulo 128 specifies 0 - 127. Default is 8.

Note: When you change this value, the transmit and receive window sizes become invalid. Use the **set station** command to change the receive-window and transmit-window sizes.

Example: `set link modulo 8`

link name
Dynamically changes the name of the link without affecting the SRAM configuration. A maximum of 8 characters can be entered. This parameter is for informational purposes only.

Example: `set link name`

```
Enter link name: [LINK_0]?
```

link poll delay or timeout or retry
Dynamically changes the following poll information without affecting the SRAM configuration.

delay Configures the delay between each poll that is sent over the interface.

timeout Configures the amount of time the router waits for a poll response before timing out.

retry Configures the number of times the interface retries to poll the remote SDLC link station before it closes the connection.

Example: `set link poll delay`

```
Enter delay between polls [0.2]?
```

link role *primary*, *secondary*, or *negotiable*
Configures the interface as an SDLC primary, secondary, or negotiable link station. The default is primary. Use of this command does not affect the SRAM configuration. Information removed 8-14-96. LAD

Example: `set link role primary`

link rts-hold
Dynamically changes the time to hold Request to Send (RTS) high after transmitting a frame without affecting the SRAM configuration. This setting is for half-duplex mode. This setting has no effect in full-duplex mode.

Example: `set link rts-hold`

```
Enter RTS hold duration after transmit complete [0.0]?
```

link snrm *timeout* or *retry*
For primary stations, dynamically changes the following SNRM(E) information without affecting the SRAM configuration.

timeout The time to wait for an Unnumbered Acknowledgment (UA) response before retransmitting an SNRM(E).

retry The number of times to retransmit an SNRM(E) without receiving a response before giving up.

Example: `set link snrm timeout`

```
Enter SNRM response timeout [2.0]?
```

link type multipoint or point-to-point
Dynamically changes the SDLC link to either a multipoint link or a point-to-point link without affecting the SRAM configuration.

Example: `set link type multipoint`

link xid/test *timeout* or *retry*
For primary stations, dynamically changes the following XID/test information without affecting the SRAM configuration.

timeout The maximum amount of time to wait for an XID or TEST frame response before retransmitting the test frame.

retry The maximum number of times an XID or TEST frame is resent before giving up. A 0 (zero) causes the router to retry indefinitely.

Example: `set link xid timeout 10`

Note: Examples for, and explanations of, the following parameters can be found in the SDLC configuration chapter at"Set" on page 41-6.

station *address* or *name* address
Changes the station's SDLC address.

station *address* or *name* max-packet
Maximum size of packet that this station can receive.

station *address* or *name* name
Name of the SDLC station.

station *address* or *name* receive-window
Maximum number of frames router sends before responding.

station *address* or *name* transmit-window
Maximum number of frames router transmits before receiving a response frame.

Test

Transmits a specified number of TEST frames to the specified station and waits for a response. Use this command to test the integrity of the connection. Press any key to cancel the test.

Note: Disable the specified link station before using this command.

Syntax: test station *name* or *address #frames-to-send frame-size*

Example: `test station c1`

```
Number of frames to send [1]? 5
Frame length [265]?
Starting echo test -- press any key to abort
5 frames sent, 5 frames received, 0 compare errors, 0 timeouts
```

Number of test frames to send
Total number of frames to send.

Frame length Length of frames to be sent. Frame length cannot be larger than the maximum frame length of the specified station.

The test may be aborted by pressing any key.

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

SDLC Interfaces and the GWCON Interface Command

While the SDLC interface has a console process for monitoring purposes, the 2210 also displays complete statistics for installed interfaces when you use the **interface** command from the GWCON environment. (For more information on the interface command, refer to Chapter 6, “The GWCON (Monitoring) Process and Commands” on page 6-1.)

Statistics Displayed for SDLC Interfaces

Using the **interface** command, you can display statistics for SDLC devices without entering the SDLC monitoring process. To do this, enter the **interface** command and an interface number at the + prompt, as shown:

Nt Indicates the interface number as assigned by software during initial configuration.

Nt' Indicates the interface number as assigned by software during initial configuration.

Note: For SDLC interfaces, the Nt' interface number is always the same as the Nt interface number.

CSR Indicates the memory location of the control status register for the SDLC interface.

Self-test passed Indicates the total number of times the SDLC interface passed its self-test.

Self-test failed Indicates the total number of times the SDLC interface was unable pass its self-test.

Maintenance failed Indicates the number of maintenance failures.

The following parameters are displayed only if a cable is connected. The information displayed depends on the cable that is connected. Different parameters are displayed with other cables.

Adapter cable Indicates the type of adapter cable that the level converter is using.

V.24 circuit Indicates the circuits being used on the V.24.

Nicknames Indicates the signals being used on the V.24 circuit.

RS-232 The EIA 232 (RS 232) circuit names.

State Indicates the state of V.24 circuits, signals, and pin assignments (ON or OFF).

Line speed (configured) Indicates the currently configured line speed for the SDLC interface.

Last port reset Indicates how long ago the port was last reset.

Input frame errors Indicates the input frame error type (CRC error, too short, aborted, alignment, too long, DMA/FIFO overrun) and the total number of errors that have occurred.

Output frame counters
: Indicates the total number of DMA/FIFO overruns and output aborts sent for output frames.

Missed frame
: When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.

L & F bits not set
: On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the Last and First bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the `L & F bits not set` counter is incremented, and the buffer is cleared for reuse.

  Note: It is unlikely that the `L & F bits not set` counter will be affected by traffic.

Chapter 43. Using and Configuring the V.25bis Network Interface

The V.25bis interface allows routers to establish serial connections over switched telephone lines using V.25bis modems. This chapter describes how to configure a V.25bis interface. It includes the following sections:

Note: You can assign a destination name to a **connection list** and assign a destination number to each line in the list. When that destination name is called, the numbers in the list are tried one by one until a connection is made or the list is exhausted.

Accessing the Interface Configuration Process

Use the following procedure to access the V.25bis configuration process.

1. At the OPCON prompt (*), enter the **status** command to find the PID for CONFIG. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the **talk** command and the PID for CONFIG. (For more detail on this command, refer to Chapter 2, The OPCON Process and Commands.) For example:

   ```
   * talk 6
   Config>
   ```

   After you enter the **talk 6** command, the CONFIG prompt (`Config>`) displays on the console. If the prompt does not appear when you first enter **CONFIG**, press **Return** again.
3. At the CONFIG prompt, enter the **list devices** command to display the network interface numbers for which the router is currently configured. For example:

   ```
   Config> list devices

   Ifc 0 Ethernet                      CSR  81600, CSR2  80C00, vector 94
   Ifc 1 V.25bis                       CSR  81620, CSR2  80D00, vector 93
   Ifc 2 WAN X.25                      CSR  81640, CSR2  80E00, vector 92
   Ifc 3 WAN PPP                       CSR 381620, CSR2 380D00, vector 125
   Ifc 4 WAN Frame Relay               CSR 381640, CSR2 380E00, vector 124
   Ifc 5 Token Ring                    CSR 600000, vector 95
   ```

4. Record the interface numbers.
5. Enter the CONFIG **network** command and the number of the interface you want to configure. For example:

   ```
   Config> network 1
   V.25bis Config>
   ```

   The V.25bis configuration prompt now displays on the console.

Before You Begin

Before you configure V.25bis on the router, make sure you have the following:

- V.25bis modems that support synchronous V.25bis commands and the 1988 ITU/CCITT V.25bis specification.
- If your modem does not automatically detect answer originate, you must:
  - Configure the modem at one end of the link to originate calls.
  - Configure the modem at the other end of the link to answer calls.
  - Set up the modem on the answering end to auto-answer.

Configuration Procedures

This section describes how to configure your router for V.25bis. The tasks you need to perform are:

1. Adding V.25bis addresses
2. Configuring V.25bis parameters
3. Adding dial circuits
4. Configuring dial circuits

Note: You must restart the router for changes to the V.25bis configuration to take effect.

Adding V.25bis Addresses

You need to add a V.25bis address for each local V.25bis interface as well as for each destination. The V.25bis address includes:

- *Address Name.* The address name is a description of the address. You can use any string of up to 23 printable ASCII characters.
- *Network Dial Address.* Telephone number of the local or destination port. You can enter up to 31 characters that are in the valid format of the connected V.25bis modem.

  Note: The valid character set for telephone numbers as defined by the CCITT and supported by the IBM 2210 includes:

  - The decimal digits 0 through 9
  - Colon (:) — "Wait Tone"
  - Left-angled bracket (<) —"Pause", used for inserting a fixed delay (dependant on modem) between digit sequences. For example, when going through a PBX or PTN.
  - Equal (=) — "Separator 3", which is "for national use." (Consult your modem manual.)
  - The letter P — "Dialing to be continued in Pulse mode." (Not supported by some modems.)
  - The letter T — "Dialing to be continued in DTMF mode." (Not supported by some modems.)

To add a V.25bis address, enter the **add v25-bis-address** command at the Config> prompt. For example:

```
Config>add v25-bis-address
Assign address name [1-23] chars []? remote-site-baltimore
Assign network dial address [1-20 digits][]? 19095551234
```

Configuring the V.25bis Interface

This section explains how to configure the V.25bis interface. To configure, do the following:

1. To set up a serial line interface for V.25bis, set the data-link protocol for the serial line interface. From the Config> prompt, use the **set data-link v25bis** command. For example:

   ```
   Config>set data-link v25bis
   Interface Number [0]? 2
   ```

2. Display the V.25bis Config> prompt by entering the **network** command followed by the number of the interface. For example:

   ```
   Config>network 2
   V.25bis Data Link Configuration
   V25bis Config>
   ```

   You can use the **list devices** command at the Config> prompt to display a list of interface numbers configured on the router.

3. Use the **set local-address** command to specify the network address name of the local port. You must enter one of the address names you defined using the **add v25bis-address** command. For example:

   ```
   V25bis Config>set local-address
   Local network address name  []? remote-site-baltimore
   ```

Note: You must restart the router for configuration changes to take effect.

Optional V.25bis Parameters

The following are optional V.25bis parameters you can set. For a complete description of these commands, see “V.25bis Configuration Commands” on page 43-5.

- You can limit the number of successive calls to an address that is inaccessible or that refuses those calls. To do so, use the **set retries-no-address** and the **set timeout-no-answer** commands.
- The **set disconnect-timeout** command controls the amount of time the router waits to initiate a call after dropping a signal from the previous call.
- The **set command-delay-timeout** command specifies the amount of time the router waits to initiate or answer a call after it turns on DTR.
- The **set connect-timeout** command specifies the number of seconds allowed for a call to be established.
- When you have finished configuring the interface, you can use the **list** command to display your configuration.

Adding Dial Circuits

Dial circuits are mapped to V.25bis serial line interfaces. You can map multiple dial circuits to one serial line interface.

To add a dial circuit, use the **add device dial-circuit** command from the `Config>` prompt. The software assigns an interface number to each circuit. You will use this number to configure the dial circuit.

Example:

```
Config>add device dial-circuit
Adding device as interface 6
```

Note: Dial circuits default to the Point-to-Point protocol (PPP). You can also set the dial circuit to use Frame Relay (FR).

Configuring Dial Circuits

This section describes how to configure a dial circuit. For a complete description of the dial circuit commands, see Chapter 49, “Configuring Dial Circuits” on page 49-1. To configure the dial circuit, do the following:

1. Display the `Circuit Config>` prompt by entering the **network** command followed by the interface number of the dial circuit. You can use the **list devices** command at the `Config>` prompt to display a list of the dial circuits that you added. For example:

   ```
   Config>network 6
   Circuit configuration
   Circuit Config>
   ```

2. Map the dial circuit to a V.25bis interface. The Base net is the V.25 bis interface number. For example:

   ```
   Circuit Config>set net
   Base net for this circuit [0]?  0
   ```

3. Specify the address name of the remote router to which the dial circuit will connect. You must use one of the names you defined using the **add v25-bis-address** command. For example:

   ```
   Circuit Config>set destination
   Assign destination address name []? newyork
   ```

4. Configure the dial circuit to initiate outbound calls only, accept inbound calls only, or both initiate and accept calls.

   Use the **set calls** command. To avoid a conflict if both ends of the link attempt to establish a call at the same time, configure the dial circuit at one end of the link to accept inbound calls only, and configure the dial circuit at the other end of the link to initiate outbound calls only. For example:

   ```
   Circuit Config>set calls outbound
   Circuit Config>set calls inbound
   ```

   Note: For WAN-Restoral operations or another dial-on-demand application, you should set up the circuit for either inbound or outbound calls.

5. Specify the timeout period for the circuit.

   Use the **set idle** command. If there is no traffic over the circuit for this specified time period, the dial circuit hangs up. To configure the circuit as a dedicated circuit, set the idle timer to zero. To configure the circuit to dial on demand, set the idle timer to a value other than zero. The range is 0 to 65535 and the default is 60 seconds. For example:

```
Circuit Config>set idle
Idle timer (seconds, 0 means always active) [60]? 0
```

Note: For WAN-Restoral operations you must set the idle time to 0.

6. Optionally, you can delay the time between when a call is established and the initial packet is sent.

   Use the **set selftest-delay** command. Setting a selftest delay can prevent initial packets from being dropped. If your modems take extra time to synchronize, adjust this delay. For example:

   ```
   Circuit Config>set selftest-delay
   Selftest delay(milli-seconds,0 means no delay)[150]?200
   ```

7. Set the inbound address name.

   Use the **set inbound** command. You need to use this command only if you set up the circuit for both inbound and outbound calls and if the router's destination address is different from the destination address that the remote router dials. For example, the numbers would be different if one of the routers must go through a PBX, international, or inter-LATA exchange. For example:

   ```
   Circuit Config>set inbound
   Assign destination inbound address name []? newyork
   ```

   The inbound address name must match one of the names that you defined using the **add v25-bis-address** command.

8. Optionally, you can enter the configuration process for the data-link layer protocol that is running on the dial circuit (PPP or Frame Relay). Use the **encapsulator** command. For example:

   ```
   Circuit Config>encapsulator
   ```

V.25bis Configuration Commands

Table 43-1 summarizes and the rest of the section explains the V.25bis configuration commands. These commands allow you to display, create, or modify a V.25bis configuration. Enter the V.25bis configuration commands at the `V.25bis Config>` prompt.

Table 43-1. V.25 bis Configuration Commands Summary

Command	Function
? (Help)	Lists the configuration commands or lists the options associated with that command.
List	Displays the V.25bis configuration.
Set	Sets the local address, connect, disconnect, and no answer timeouts, number of retries after no answer, and command delay timeout.
Exit	Exits the V.25bis configuration process and returns to the `Config>` prompt.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
LIST
SET
EXIT
```

Example: **Set ?**

```
COMMAND-DELAY-TIMEOUT
CONNECT-TIMEOUT
DISCONNECT-TIMEOUT
HDLC
LOCAL-ADDRESS
RETRIES-NO-ANSWER
TIMEOUT-NO-ANSWER
```

List

Use the **list** command to display the current V.25 bis configuration.

Syntax: list

Example: **list**

```
          V.25bis Configuration

Local Network Address Name    = v403
Local Network Address         = 15088982403

Non-Responding addresses:
Retries             = 1
Timeout             = 0 seconds

Call timeouts:
Command Delay       = 0 ms
Connect             = 60 seconds
Disconnect          = 2 seconds

Cable type          = V.35 DTE
Speed               = 9600
```

Local Network Address Name:	Displays the network address name of the local port.
Local Network Address:	Displays the network dial address of the local port.
Non-responding addresses:	
Retries	Maximum number of calls the router attempts to make to a non-responding address during the timeout period.
Timeout	If the router reaches the maximum number of retries to a non-responding address, it does not attempt to establish the call until this time has expired. This timeout period begins when the router attempts the first call.
Call timeouts:	Number of call timeouts.

Command Delay	Amount of time, in milliseconds, that the router waits to initiate or answer a call after it turns on DTR (Data Terminal Ready). If you set this parameter to 0, the router waits for the modem to respond to DTR with the CTS (Clear to Send) signal before it issues commands.
Connect	Number of seconds allowed for a call to be established. If this parameter is set to 0, the modem controls the connection establishment timeout.
Disconnect	After the routers drops DTR it waits this amount of time before it initiates further calls. If you set this parameter to 0, the router waits for the modem to respond to the DTR drop by dropping CTS and DSR before it initiates the next call.

Set

Use the **set** command to configure local addresses, timeouts and delays for calls, retries and timeouts for non-responding addresses, and the HDLC cable type.

Syntax: set command-delay timeout . . .
connect-timeout . . .
hdlc cable . . .
hdlc speed . . .
disconnect-timeout . . .
local-address . . .
retries-no-answer . . .
timeout-no-answer . . .

command-delay-timeout *# of milliseconds*

After the router turns on DTR (Data Terminal Ready), it waits this amount of time before it initiates or answers a call. If you set this parameter to 0, the router waits for the modem to respond to DTR with the CTS (Clear to Send) signal before it issues commands. The range is 0 to 65535 milliseconds, and the default is 0.

Example: `set command-delay-timeout 0`

connect-timeout *# of seconds*

Sets the number of seconds allowed for a call to be established. The range is 0 to 65535 seconds, and the default is 60. If you set this parameter to 0, the modem controls the connection timeout. You should initially set this parameter to 0 and then use ELS event V25B.027 to find out how long it takes to establish connections to various destinations. You can then set this parameter to a number slightly higher than the longest connect time.

Note: Normally government regulation limits modem manufacturers to a maximum length for call setup. This value is merely an optimization, although inter-operation with some DSUs may require that you change this parameter.

Example: `set connect-timeout 10`

disconnect-timeout *# of seconds*

Specifies the amount of time, in seconds, that the router waits after dropping DTR before it initiates further calls. The range is 0 to 65535 seconds, and the default is 2. If you set this parameter to 0, the router waits for the modem to respond to the DTR drop by dropping CTS and DSR before it initiates the next call.

Example: `set disconnect-timeout 500`

hdlc cable *rs232 dte*

Specifies the type of cable connected to this interface. Setting this parameter allows you to view the cable type when you enter the **interface** command at the GWCON (+) prompt and when you enter the **statistics** command at the `V.25bis>` monitoring prompt. This parameter does not affect operation of the router.

Example: `set hdlc cable rs-232 dte`

hdlc speed

Specifies the line speed for this interface. Setting this parameter allows you to view the line speed when you enter the interface command at the GWCON (+) prompt and when you enter the statistics command at the `V.25bis>` monitoring prompt. The range is 300 to 2 048 000 bps.

Note: This command does not affect the actual line speed but it sets the speed some protocols, such as IPX, use when calculating routing cost parameters for dial circuits mapped to the V.25bis interface.

Example: `set hdlc speed 2400`

local-address *address name*

Specifies the network address name of the local port. This address name must match one of the names that you defined at the `Config>` using the **add v25-bis-address** command.

Example: `set local-address line-1-local`

retries-no-answer *value*

Some telephone service providers impose restrictions on automatic recalling devices to limit the number of successive calls to an address that is inaccessible or that refuses those calls. This parameter specifies the maximum number of calls the router attempts to make to a non-responding address during the timeout period. The range is 0 to 10, and the default is 1.

Note: Government regulation may also impose limits on the modem manufacturer that would supersede this parameter.

Example: `set retries-no-answer 2`

timeout-no-answer *# of seconds*

After the router reaches the maximum number of **retries-no-answer** to a non-responding address, it does not initiate further calls to that address until this time has expired. This timeout period begins when the router attempts the first call to an address. The range is 0 to 65535 seconds, and the default is 0. If you set this parameter to 0, the modem controls the timeout period.

Example: `set timeout-no-answer 180`

Exit

Use the **exit** command to return to the `Config>` prompt.

Syntax: exit

Example: `exit`

Chapter 44. Monitoring the V.25bis Network Interface

This chapter describes the V.25bis console commands and GWCON commands. It includes the following sections:

Accessing the Interface Console Process

To access the interface console process for V.25bis, enter the following command at the GWCON (+) prompt:

```
+ network #
```

Where # is the number of the V.25bis serial line. You cannot directly access the V.25bis console process for dial circuits, but you can monitor the dial circuits that are mapped to the serial line interface.

Note: V.25bis interfaces also have ELS troubleshooting messages that you can use to monitor V.25bis related activity. See the *IBM Nways Event Logging System Messages Guide* for further details.

V.25bis Console Commands

This section summarizes and explains the V.25bis console commands. These commands allow you to view the calls, circuits, parameters, and statistics of the V.25bis interfaces.

Enter the V.25bis console commands at the `V.25bis>` prompt.

Table 44-1. V.25bis Console Command Summary

Console Command	Function
? (Help)	Lists the V.25bis console commands or lists the options associated with specific commands.
Calls	List the number of completed and attempted connections made for each dial circuit mapped to this interface since the last time statistics were reset on the router.
Circuits	Shows the status of all data circuits configured on the V.25 bis interface.
Parameters	Displays the current parameters for the V.25bis interface. (This command is similar to the `V.25bis Config>` list command.)
Statistics	Displays the current statistics for the V.25bis interface.
Exit	Exits the V.25bis console process and returns to the GWCON (+) process.

? (Help)

Use the **?** (help) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
CALLS
CIRCUITS
PARAMETERS
STATISTICS
EXIT
```

Calls

Use the **calls** command to list the number of completed and attempted connections made for each dial circuit mapped to this interface since the last time statistics were reset on the router.

Syntax: calls

Example: **calls**

```
Net Interface  Site Name           In    Out   Rfsd  Blckd
1     PPP/0     v403               2      0      0      0

Unmapped connection indications:    0
```

Net	Number of the dial circuit mapped to this interface.
Interface	Type of interface and its instance number.
Site Name	Network address name of the dial circuit.
In	Number of inbound connections accepted for this dial circuit.
Out	Number of completed connections initiated by this dial circuit.
Rfsd	Number of connections initiated by this dial circuit that were refused by the network or the remote destination port.
Blckd	Number of connection attempts that the router blocked. The router blocks connection attempts if the local port is already in use, the maximum number of retries to a non-responding address is reached, or a modem is not responding.
Unmapped connection indications:	Number of connection attempts that were refused by the router because there were no enabled dial circuits that were configured to accept the incoming calls.

Circuits

The **circuits** command shows the status of all dial circuits configured on the V.25bis port.

Syntax: circuits

Example: **circuit**

```
Net Interface  MAC/Data-Link    State     Reason      Duration
2   PPP/0      Point to Point   Avail     Rmt Disc     1:02:25
```

Net	Number of the dial circuit mapped to this interface
Interface	Type of interface and its instance number.
MAC/DataLink	Type of datalink protocol configured for this dial circuit.
State	Current state of the dial circuit: Up - currently connected Available - not currently connected, but is available Disabled - dial circuit was disabled Down - failed to connect because of a busy dial circuit or because the link-layer protocol is down
Reason	Reason for the current state: nnn_Data - (where nnn is the name of a protocol) the circuit is Up because a protocol had data to send. Remote Disconnect - the circuit is either Down or Available because the remote destination disconnected the call. Operator Request - the circuit is Available because the last call was disconnected by a console command. Inbound - the circuit is Up because the circuit answered an inbound call. Restoral - the circuit is Up because of a WAN Restoral operation. Self Test - the circuit was configured as static (idle time=0) and successfully connected once it was enabled.
Duration	Length of time that the circuit has been in the current state.

Parameters

Use the **parameters** command to display the current V.25bis serial line configuration. Note that this is the same information displayed in the `V.25bis Config>` list command.

Syntax: parameters

Example: **`parameters`**

```
         V.25bis port Parameters

Local Network Address Name    = v402
Local Network Address         = 15088982402

Non-Responding addresses:
Retries             = 1
Timeout             = 0 seconds

Call timeouts:
Command Delay       = 0 ms
Connect             = 0 seconds
Disconnect          = 0 seconds
```

Local Network Address Name:	Network address name of the local port.
Local Network Address:	Network dial address of the local port.
Non-responding addresses:	
Retries	Maximum number of calls the router attempts to make to a non-responding address during the timeout period.

Timeout	If the router reaches the maximum number of retries to a non-responding address, it does not attempt to establish the call until this time has expired. This timeout period begins when the router attempts the first call to an address.

Call timeouts:

Command Delay	Amount of time, in milliseconds, that the router waits to initiate or answer a call after it turns on DTR (Data Terminal Ready). If you set this parameter to 0, the router waits for the modem to respond to DTR with the CTS (Clear to Send) signal before it issues commands.
Connect	Number of seconds allowed for a call to be established. If this parameter is set to 0, the modem controls the connection establishment timeout.
Disconnect	After the routers drops DTR it waits this amount of time before it initiates further calls. If you set this parameter to 0, the router waits for the modem to respond to the DTR drop by dropping CTS and DSR before it initiates the next call.

Statistics

Use the **statistics** command to display the current statistics for this V.25bis interface.

Syntax: statistics

Example: **statistics**

```
        V.25bis port Statistics

  Adapter cable:              RS-232 DTE  RISC Microcode Revision: 1



   V.24 circuit: 105 106 107 108 109 125 141
   Nicknames:    RTS CTS DSR DTR DCD RI  LL
   RS-232        CA  CB  CC  CD  CF  CE
   State:        OFF OFF OFF OFF OFF OFF OFF



  Line speed:             4800
  Last port reset:        24 seconds ago

  Input frame errors:
   CRC error                    0   alignment (byte length)   0
   missed frame                 0   too long (>  2182 bytes)  0
   aborted frame                0   DMA/FIFO overrun          0
   L & F bits not set           0
Output frame counters:
   DMA/FIFO underrun errors     0   Output aborts sent        0
```

Adapter cable:	Type of adapter cable being used.
V.24 circuit:	Circuit numbers as identified by V.24 specifications.
Nicknames:	Common names for the circuits.
RS-232	EIA 232 (also known as RS-232) names for the circuits.
State:	Current state of the circuits: ON, OFF, or "---," which means that the state is undefined for this type of interface.

Line speed:	The transmit clock speed (approximate).
Last port reset:	Length of time since the port was reset.
Input frame errors:	
CRC error	Number of packets received that contained checksum errors and as a result were discarded.
Alignment (byte length)	Number of packets received that were not an even multiple of 8 bits in length and as a result were discarded.
Missed Frame	When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.
too long (> nnnn bytes)	Number of packets received that were greater than the configured frame size (nnnn) and as a result were discarded.
aborted frame	Number of packets received that were aborted by the sender or a line error.
DMA/FIFO overrun	The number of times the serial interface card could not send data fast enough to the system packet buffer memory to receive packets from the network.
L & F bits not set	On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the L & F bits not set counter is incremented, and the buffer is cleared for reuse. **Note:** It is unlikely that the L & F bits not set counter will be affected by traffic.
Output frame counters:	
DMA/FIFO underrun errors	Number of times the serial interface card could not retrieve data fast enough from the system packet buffer memory to transmit packets onto the network.
Output aborts sent	Number of transmissions that were aborted as requested by upper-level software.

Exit

Use the **exit** command to return to the GWCON (+) prompt.

Syntax: exit

Example: `exit`

V.25bis and the GWCON Commands

While V.25bis has its own console process for monitoring purposes, the router also displays configuration information and complete statistics for devices and circuits when you use the interface, statistics, and error commands from the GWCON environment. You can also use the GWCON **test** command to test DCEs and circuits.

Note: Issuing the **test** command to the V.25 bis serial interface causes the current call to be dropped and re-dialed.

For more information on the GWCON command, see Chapter 6, “The GWCON (Monitoring) Process and Commands” on page 6-1.

Statistics for V.25bis Interfaces and Dial Circuits

Use the **interface** command at the GWCON (+) prompt to display statistics for V.25bis serial line interfaces and dial circuits.

To display the following statistics for a V.25bis serial line interface, use the **interface** command followed by the *interface number* of the V.25bis serial line interface.

Example: `interface 1`

```
                                Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
1  1   V.25/0    80000000   44          1          0            0
V.25bis MAC/data-link on SCC Serial Line interface

Adapter cable:       RS-232 DTE       RISC Microcode Revision: 1

    V.24 circuit: 105 106 107 108 109 125
    Nicknames:    RTS CTS DSR DTR DCD R1  LL
    RS-232:       CA  CB  CC  CD  CF  CE
    State:        OFF OFF OFF OFF OFF OFF OFF

Line Speed:               14.400 Kbps
Last port reset:          1 hour, 28 minutes, 25 seconds ago

Input frame errors:
    CRC error                    0   alignment (byte length)   0
    missed frame                 0   too long (>  2182 bytes)  0
    aborted frame                0   DMA/FIFO overrun          0

    Output frame counters:     DMA/FIFO underrun errors     0   Output aborts sent         0
```

To display the following statistics for a dial circuit, use the **interface** command followed by the *interface number* of the dial circuit.

Example: `interface 3`

```
                                Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
3  2   PPP/1        81640   5C          0          5            0
  Point to Point MAC/data-link on V.25bis Dial Circuit interface
```

The following table describes the output for both serial line interfaces and dial circuits.

Nt	Serial line interface number or dial circuit interface number.
Nt′	If “Nt” is a dial circuit, this is the interface number of the V.25bis serial line interface to which the dial circuit is mapped.
Interface	Interface type and its instance number.
CSR	Command and status register addresses of base network.
Vec	Interrupt vector address.
Self-Test Passed	Number of self-tests that succeeded.
Self-Test Failed	Number of self-tests that failed.
Maintenance: Failed	Number of maintenance failures.
Adapter cable:	Type of adapter cable that is being used.

V.24 circuit:	Circuit numbers as identified by V.24 specifications.
Nicknames	Common names for the circuits.
RS-232	EIA 232 (also known as RS-232) names for the circuits.
State	Current state of the circuits (ON or OFF).
Line speed	The transmit clock speed (approximate).
Last port reset	Length of time since the port was reset.
Input frame errors:	
CRC error	Number of packets received that contained checksum errors and as a result were discarded.
Alignment (byte length)	Number of packets received that were not an even multiple of 8 bits in length and as a result were discarded.
Missed Frame	When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.
too long (> nnnn bytes)	Number of packets received that were greater than the configured frame size and as a result were discarded.
DMA/FIFO overrun	The number of times the serial interface card could not send data fast enough to the system packet buffer memory to receive packets from the network.
L & F bits not set	On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the L & F bits not set counter is incremented, and the buffer is cleared for reuse. **Note:** It is unlikely that the L & F bits not set counter will be affected by traffic.
aborted frame	Number of packets received that were aborted by the sender or a line error.
Output frame counters:	
DMA/FIFO underrun errors	Number of times the serial interface card could not retrieve data fast enough from the system packet buffer memory to transmit packets onto the network.
Output aborts sent	Number of transmissions that were aborted as requested by upper-level software.

Chapter 45. Using and Configuring the V.34 Network Interface

The V.34 interface allows routers to establish serial connections over switched telephone lines using modems that support the standard AT command set. This chapter describes how to configure a V.34 interface. It includes the following sections:

Note: You can assign a destination name to a **connection list** and assign a destination number to each line in the list. When that destination name is called, the numbers in the list are tried one by one until a connection is made or the list is exhausted.

Accessing the Interface Configuration Process

Use the following procedure to access the V.34 configuration process.

1. At the OPCON prompt (*), enter the **status** command to find the PID for CONFIG. (See page 1-5 for sample output of the **status** command.)
2. At the OPCON prompt, enter the **talk** command and the PID for CONFIG. (For more detail on this command, refer to Chapter 2, The OPCON Process and Commands.) For example:

   ```
   * talk 6
   Config>
   ```

   After you enter the **talk 6** command, the CONFIG prompt (`Config>`) displays on the console. If the prompt does not appear when you first enter **CONFIG**, press **Return** again.
3. At the CONFIG prompt, enter the **list devices** command to display the network interface numbers for which the router is currently configured. For example:

   ```
   Config> list devices
   Ifc 0 Ethernet                    CSR  81600, CSR2  80C00, vector 94
   Ifc 1 V.34 Base Net               CSR  81620, CSR2  80D00, vector 93
   Ifc 2 WAN X.25                    CSR  81640, CSR2  80E00, vector 92
   Ifc 3 WAN PPP                     CSR 381620, CSR2 380D00, vector 125
   Ifc 4 WAN Frame Relay             CSR 381640, CSR2 380E00, vector 124
   Ifc 5 Token Ring                  CSR 600000, vector 95
   ```
4. The V.34 interfaces are listed as "V.34 Base Net." Record the interface numbers of interfaces to configure.
5. Enter the CONFIG **network** command and the number of the interface you want to configure. For example:

   ```
   Config> network 1
   V.34 System Net Config >
   ```

   The V.34 configuration prompt now displays on the console.

Before You Begin

Before you configure V.34 on the router, make sure you have asynchronous modems that support the Hayes AT command set and that you know the maximum DTE speed of each modem.

Configuration Procedures

This section describes how to configure your router for V.34. The tasks you need to perform are:

1. Adding V.34 addresses
2. Configuring V.34 parameters
3. Adding dial circuits
4. Configuring dial circuits

Note: You must restart the router for changes to the V.34 configuration to take effect.

Adding V.34 Addresses

A default V.34 address is created when V.34 interfaces are initially configured (called “default_address”). Dial circuits configured on the V.34 interface default to the same address allowing some dial-in applications to work without modification of the V.34 address.

You need to add a V.34 address (or modify the default_address) if you plan to use dial-out applications. The V.34 address includes:

- *Address Name.* The address name is a description of the address. You can use any string of up to 23 printable ASCII characters.
- *Network Dial Address.* Telephone number of the local or destination port. You can enter up to 31 characters that are in the valid dial characters for the connected modem.

  Note: The valid character set for telephone numbers as defined by the CCITT and supported by the IBM 2210 includes:

  - The decimal digits 0 through 9
  - Colon (:) – "Wait Tone"
  - Left-angled bracket (<) – "Pause", used for inserting a fixed delay (dependent on modem) between digit sequences. For example, when going through a PBX or PTN.
  - Equal (=) – "Separator 3", which is "for national use." (Consult your modem manual.)
  - The letter P – "Dialing to be continued in Pulse mode." (Not supported by some modems.)
  - The letter T – "Dialing to be continued in DTMF mode." (Not supported by some modems.)

V.34 addresses are not interface specific so they are added from the main `Config>` prompt. For example:

```
Config>add v34-address
Assign address name [1-23] chars []? remote-site-baltimore
Assign network dial address [1-20 digits][]? 1-909-555-1234
```

Configuring the V.34 Interface

This section explains how to configure the V.34 interface. To configure, do the following:

1. To set up a serial line interface for V.34, set the data-link protocol for the serial line interface. From the `Config>` prompt, use the **set data-link v34** command. For example:

   ```
   Config> set data-link v34
   Interface Number [0]? 2
   ```

2. Display the `V.34 Config>` prompt by entering the **network** command followed by the number of the interface. For example:

   ```
   Config>network 2
   V.34 Data Link Configuration
   V34 System Net Config    2>
   ```

   You can use the **list devices** command at the `Config>` prompt to display a list of interface numbers configured on the router.

3. Use the **set local-address** command to specify the network address name of the local port. You must enter one of the address names you defined using the **add v34-address** command. For example:

   ```
   V34 System Net Config   2>set local-address
   Local network address name  []? remote-site-baltimore
   ```

Note: You must restart the router for configuration changes to take effect.

Optional V.34 Parameters

The following are optional V.34 parameters you can set. For a complete description of these commands, see “V.34 Configuration Commands” on page 45-5.

- You can limit the number of successive calls to an address that is inaccessible or that refuses those calls. To do so, use the **set retries-no-address** and the **set timeout-no-answer** commands.
- The **set disconnect-timeout** command controls the amount of time the router waits to initiate a call after dropping a signal from the previous call.
- The **set command-delay-timeout** command specifies the amount of time the router waits to initiate or answer a call after it turns on DTR.
- The **set connect-timeout** command specifies the number of seconds allowed for a call to be established.
- The **speed** command sets the maximum DTE speed for the modem.
- The **modem-init-string** command allows flexibility in modem configuration to accommodate user or external equipment requirements.
- When you have finished configuring the interface, you can use the **list** command to display your configuration.

Adding Dial Circuits

Dial circuits are mapped to V.34 serial line interfaces. You can map multiple dial circuits to one serial line interface.

The V.34 interface supports multiple types of dial circuits. To add a dial circuit use one of the following commands from the Config> prompt.

- **add device dial-circuit**
- **add device dial-in**
- **add device dial-out**

The software assigns an interface number to each circuit. You will use this number to configure the dial circuit.

Example:

```
Config> add device dial-circuit
Adding device as interface 6
```

Note: Dial circuits default to the Point-to-Point protocol (PPP). Although the set data-link command can be used to set the data-link of a dial circuit to Frame Relay, only PPP dial circuits are supported over V.34.

Configuring Dial Circuits

This section describes how to configure a dial circuit. For a complete description of the dial circuit commands, see Chapter 49, “Configuring Dial Circuits” on page 49-1. To configure the dial circuit, do the following:

1. Display the Circuit Config> prompt by entering the **network** command followed by the interface number of the dial circuit. You can use the **list devices** command at the Config> prompt to display a list of the dial circuits that you added. For example:

   ```
   Config>network 6
   Circuit configuration
   Circuit Config>
   ```

2. Map the dial circuit to a V.34 interface. The Base net is the V.34 interface number. For example:

   ```
   Circuit Config>set net
   Base net for this circuit [0]?  0
   ```

3. Specify the address name of the remote router to which the dial circuit will connect. You must use one of the names you defined using the **add v34-address** command. For example:

   ```
   Circuit Config>set destination
   Assign destination address name []? newyork
   ```

4. Configure the dial circuit to initiate outbound calls only, accept inbound calls only, or both initiate and accept calls.

   Use the **set calls** command. To avoid a conflict if both ends of the link attempt to establish a call at the same time, configure the dial circuit at one end of the link to accept inbound calls only, and configure the dial circuit at the other end of the link to initiate outbound calls only. For example:

   ```
   Circuit Config>set calls outbound
   Circuit Config>set calls inbound
   ```

   Note: For WAN-Restoral operations or another dial-on-demand application, you should set up the circuit for either inbound or outbound calls.

5. Specify the timeout period for the circuit.

   Use the **set idle** command. If there is no traffic over the circuit for this specified time period, the dial circuit hangs up. To configure the circuit as a dedicated circuit, set the idle timer to zero. To configure the circuit to dial on demand, set the idle timer to a value other than zero. The range is 0 to 65535 and the default is 60 seconds. For example:

   ```
   Circuit Config>set idle
   Idle timer (seconds, 0 means always active) [60]? 0
   ```

   Note: For WAN-Restoral operations you must set the idle time to 0.

6. Optionally, you can delay the time between when a call is established and the initial packet is sent.

   Use the **set selftest-delay** command. Setting a self-test delay can prevent initial packets from being dropped. If your modems take extra time to synchronize, adjust this delay. For example:

   ```
   Circuit Config>set selftest-delay
   Selftest delay(milli-seconds,0 means no delay)[150]?200
   ```

7. Set the inbound address name.

   Use the **set inbound** command. You need to use this command only if you set up the circuit for both inbound and outbound calls and if the router's destination address is different from the destination address that the remote router dials. For example, the numbers would be different if one of the routers must go through a PBX, international, or inter-LATA exchange. For example:

   ```
   Circuit Config>set inbound
   Assign destination inbound address name []? newyork
   ```

   The inbound address name must match one of the names that you defined using the **add v34-address** command.

8. Optionally, you can enter the configuration process for the data-link layer protocol that is running on the dial circuit (PPP or Frame Relay). Use the **encapsulator** command. For example:

   ```
   Circuit Config>encapsulator
   ```

V.34 Configuration Commands

Table 45-1 summarizes and the rest of the section explains the V.34 configuration commands. These commands allow you to display, create, or modify a V.34 configuration. Enter the V.34 configuration commands at the `V.34 Config>` prompt.

Table 45-1. V.34 Configuration Commands Summary

Command	Function
? (Help)	Lists the configuration commands or lists the options associated with that command.
List	Displays the V.34 configuration.
Set	Sets the local address, connect, disconnect, and no answer timeouts, number of retries after no answer, and command delay timeout.
Exit	Exits the V.34 configuration process and returns to the `Config>` prompt.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
LIST
SET
EXIT
```

Example: **Set ?**

```
COMMAND-DELAY-TIMEOUT
CONNECT-TIMEOUT
DISCONNECT-TIMEOUT
SPEED
LOCAL-ADDRESS
MODEM-INIT-STRING
RETRIES-NO-ANSWER
TIMEOUT-NO-ANSWER
```

List

Use the **list** command to display the current V.34 configuration.

Syntax: list

Example: **list**

```
                V.34 System Net Configuration:

Local Network Address Name    = v403
Local Network Address         = 1-508-898-2403

Non-Responding addresses:
Retries                       = 1
Timeout                       = 0 seconds

Call timeouts:
Command Delay                 = 0 ms
Connect                       = 60 seconds
Disconnect                    = 2 seconds

Modem strings:
Initialization string         = at&f&s1l1&d2&c1x3

Speed (bps)                   = 115200
```

Local Network Address Name:	Displays the network address name of the local port.
Local Network Address:	Displays the network dial address of the local port.
Non-responding addresses:	
Retries	Maximum number of calls the router attempts to make to a non-responding address during the timeout period.
Timeout	If the router reaches the maximum number of retries to a non-responding address, it does not attempt to establish the call until this time has expired. This timeout period begins when the router attempts the first call.
Call timeouts:	Number of call timeouts.

Command Delay	Amount of time, in milliseconds, that the router waits to initiate or answer a call after it turns on DTR (Data Terminal Ready). If you set this parameter to 0, the router waits for the modem to respond to DTR with the CTS (Clear to Send) signal before it issues commands.
Connect	Number of seconds allowed for a call to be established. If this parameter is set to 0, the modem controls the connection establishment timeout.
Disconnect	After the routers drops DTR it waits this amount of time before it initiates further calls. If you set this parameter to 0, the router waits for the modem to respond to the DTR drop by dropping CTS and DSR before it initiates the next call.
Modem strings:	Command strings sent to the attached modem.
Initialization string	This is the last AT command string sent to the modem during initialization (before a call is accepted or attempted). A default string is provided which should work for most modems.
Speed (bps)	This is the DTE speed. The default should work for most modems, but you may need to set the speed lower to operate properly or higher to achieve maximum data speeds supported by the modem.

Set

Use the **set** command to configure local addresses, timeouts and delays for calls, retries and timeouts for non-responding addresses, and the HDLC cable type.

Syntax: set command-delay timeout . . .
connect-timeout . . .
disconnect-timeout . . .
speed . . .
local-address . . .
modem-init-string . . .
retries-no-answer . . .
timeout-no-answer . . .

command-delay-timeout *# of milliseconds*

After the router turns on DTR (Data Terminal Ready), it waits this amount of time before it initiates or answers a call. If you set this parameter to 0, the router waits for the modem to respond to DTR with the CTS (Clear to Send) signal before it issues commands. The range is 0 to 65535 milliseconds, and the default is 0.

Example: `set command-delay-timeout 0`

connect-timeout *# of seconds*

Sets the number of seconds allowed for a call to be established. The range is 0 to 65535 seconds, and the default is 60. If you set this parameter to 0, the modem controls the connection timeout. You should initially set this parameter to 0 and then use ELS event V34B.027 to find out how long it takes to establish connections to various destinations. You can then set this parameter to a number slightly higher than the longest connect time.

Note: Normally government regulation limits modem manufacturers to a maximum length for call setup. This value is merely an optimization, although inter-operation with some DSUs may require that you change this parameter.

Example: `set connect-timeout 10`

disconnect-timeout *# of seconds*

Specifies the amount of time, in seconds, that the router waits after dropping DTR before it initiates further calls. The range is 0 to 65535 seconds, and the default is 2. If you set this parameter to 0, the router waits for the modem to respond to the DTR drop by dropping CTS and DSR before it initiates the next call.

speed *# bits per second*

Specifies the DTE speed in bits per second for the modem. You should try to use the maximum speed supported by the modem, although some modems may not autobaud properly at all supported speeds. If you suspect there is a problem, try a lower speed.

Example: `set speed 57600`

local-address *address name*

Specifies the network address name of the local port. This address name must match one of the names that you defined at the `Config>` using the **add v34-address** command.

Example: `set local-address line-1-local`

modem-init-string *value*

This is an AT command string sent to the modem at the end of successful interface initialization. It can be used to tailor modem parameters for your application.

Example: `set modem-init at&f&s1l1&d2&c1x3`

retries-no-answer *value*

Some telephone service providers impose restrictions on automatic recalling devices to limit the number of successive calls to an address that is inaccessible or that refuses those calls. This parameter specifies the maximum number of calls the router attempts to make to a non-responding address during the timeout period. The range is 0 to 10, and the default is 1.

Note: Government regulation may also impose limits on the modem manufacturer that would supersede this parameter.

Example: `set retries-no-answer 2`

timeout-no-answer *# of seconds*

After the router reaches the maximum number of **retries-no-answer** to a non-responding address, it does not initiate further calls to that address until this time has expired. This timeout period begins when the router attempts the first call to an address. The range is 0 to 65535 seconds, and the default is 0. If you set this parameter to 0, the modem controls the timeout period.

Example: `set timeout-no-answer 180`

Exit

Use the **exit** command to return to the `Config>` prompt.

Syntax: exit

Example: `exit`

Chapter 46. Monitoring the V.34 Network Interface

This chapter describes the V.34 console commands and GWCON commands. It includes the following sections:

- "Accessing the Interface Console Process"
- "V.34 Console Commands"
- "V.34 and the GWCON Commands" on page 46-5

Accessing the Interface Console Process

To access the interface console process for V.34, enter the following command at the GWCON (+) prompt:

```
+ network #
```

Where # is the number of the V.34 interface. You cannot directly access the V.34 console process for dial circuits, but you can monitor the dial circuits that are mapped to the serial line interface.

Note: V.34 interfaces also have ELS troubleshooting messages that you can use to monitor V.34 related activity. See the *IBM Nways Event Logging System Messages Guide* for further details.

V.34 Console Commands

This section summarizes and explains the V.34 console commands. These commands allow you to view the calls, circuits, parameters, and statistics of the V.34 interfaces.

Enter the V.34 console commands at the `V.34>` prompt.

Table 46-1. V.34 Console Command Summary

Console Command	Function
? (Help)	Lists the V.34 console commands or lists the options associated with specific commands.
Calls	List the number of completed and attempted connections made for each dial circuit mapped to this interface since the last time statistics were reset on the router.
Circuits	Shows the status of all data circuits configured on the V.34 interface.
Reset	Clears connections and resets the interface.
Parameters	Displays the current parameters for the V.34 interface. (This command displays the same information as the interface configuration "list" command.)
Statistics	Displays the current statistics for the V.34 interface.
Exit	Exits the V.34 console process and returns to the GWCON (+) process.

? (Help)

Use the **?** (help) command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **?**

```
CALLS
CIRCUITS
RESET
PARAMETERS
STATISTICS
EXIT
```

Calls

Use the **calls** command to list the number of completed and attempted connections made for each dial circuit mapped to this interface since the last time statistics were reset on the router.

Syntax: calls

Example: **calls**

```
Net Interface  Site Name           In    Out   Rfsd  Blckd
1     PPP/0     v403               2      0      0      0

Unmapped connection indications:    0
```

Net	Number of the dial circuit mapped to this interface.
Interface	Type of interface and its instance number.
Site Name	Network address name of the dial circuit.
In	Number of inbound connections accepted for this dial circuit.
Out	Number of completed connections initiated by this dial circuit.
Rfsd	Number of connections initiated by this dial circuit that were refused by the network or the remote destination port.
Blckd	Number of connection attempts that the router blocked. The router blocks connection attempts if the local port is already in use, the maximum number of retries to a non-responding address is reached, or a modem is not responding.
Unmapped connection indications:	Number of connection attempts that were refused by the router because there were no enabled dial circuits that were configured to accept the incoming calls.

Circuits

The **circuits** command shows the status of all dial circuits configured on the V.34 port.

Syntax: circuits

Example: **circuit**

```
Net Interface  MAC/Data-Link    State     Reason      Duration
2   PPP/0      Point to Point   Avail     Rmt Disc     1:02:25
```

Net	Number of the dial circuit mapped to this interface
Interface	Type of interface and its instance number.
MAC/DataLink	Type of datalink protocol configured for this dial circuit.
State	Current state of the dial circuit: Up - currently connected Available - not currently connected, but is available Disabled - dial circuit was disabled Down - failed to connect because of a busy dial circuit or because the link-layer protocol is down
Reason	Reason for the current state: nnn_Data - (where nnn is the name of a protocol) the circuit is Up because a protocol had data to send. Remote Disconnect - the circuit is either Down or Available because the remote destination disconnected the call. Operator Request - the circuit is Available because the last call was disconnected by a console command. Inbound - the circuit is Up because the circuit answered an inbound call. Restoral - the circuit is Up because of a WAN Restoral operation. Self Test - the circuit was configured as static (idle time=0) and successfully connected once it was enabled.
Duration	Length of time that the circuit has been in the current state.

Parameters

Use the **parameters** command to display the current V.34 serial line configuration. Note that this is the same information displayed in the `V.34 Config>` list command.

Syntax: parameters

Example: **`parameters`**

```
        V.34 port Parameters

Local Network Address Name    = v402
Local Network Address         = 1-508-898-2402

Non-Responding addresses:
Retries                       = 1
Timeout                       = 0 seconds

Call timeouts:
Command Delay                 = 0 ms
Connect                       = 0 seconds
Disconnect                    = 0 seconds

Modem strings:
Initialization string         = at&f&s1l1&c1x3
```

Local Network Address Name:	Network address name of the local port.
Local Network Address:	Network dial address of the local port.

Non-responding addresses:

Retries	Maximum number of calls the router attempts to make to a non-responding address during the timeout period.
Timeout	If the router reaches the maximum number of retries to a non-responding address, it does not attempt to establish the call until this time has expired. This timeout period begins when the router attempts the first call to an address.

Call timeouts:

Command Delay	Amount of time, in milliseconds, that the router waits to initiate or answer a call after it turns on DTR (Data Terminal Ready). If you set this parameter to 0, the router waits for the modem to respond to DTR with the CTS (Clear to Send) signal before it issues commands.
Connect	Number of seconds allowed for a call to be established. If this parameter is set to 0, the modem controls the connection establishment timeout.
Disconnect	After the routers drops DTR it waits this amount of time before it initiates further calls. If you set this parameter to 0, the router waits for the modem to respond to the DTR drop by dropping CTS and DSR before it initiates the next call.

Statistics

Use the **statistics** command to display the current statistics for this V.34 interface.

Syntax: statistics

Example: **`statistics`**

```
        V.34 port Statistics
 Adapter cable:              RS-232 DTE  RISC Microcode Revision: 1

  V.24 circuit: 105 106 107 108 109 125 141

   Nicknames:    RTS CTS DSR DTR DCD RI  LL
   RS-232        CA  CB  CC  CD  CF  CE
   State:        OFF OFF OFF OFF OFF OFF OFF
  Line speed:             115.200 Kbps
  Last port reset:        24 seconds ago

Input frame errors:
   CRC error                    0   alignment (byte length)   0
   missed frame                 0   too long (>  2182 bytes)  0
   aborted frame                0   DMA/FIFO overrun          0
   L & F bits not set           0

Output frame counters:
   DMA/FIFO underrun errors     0   Output aborts sent        0
```

Adapter cable:	Type of adapter cable being used.
V.24 circuit:	Circuit numbers as identified by V.24 specifications.
Nicknames:	Common names for the circuits.
RS-232	EIA 232 (also known as RS-232) names for the circuits.
State:	Current state of the circuits: ON, OFF, or "---," which means that the state is undefined for this type of interface.
Line speed:	The transmit clock speed (approximate).
Last port reset:	Length of time since the port was reset.

Input frame errors:

CRC error	Number of packets received that contained checksum errors and as a result were discarded.
Alignment (byte length)	Number of packets received that were not an even multiple of 8 bits in length and as a result were discarded.
Missed Frame	When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.
too long (> nnnn bytes)	Number of packets received that were greater than the configured frame size (nnnn) and as a result were discarded.
aborted frame	Number of packets received that were aborted by the sender or a line error.
DMA/FIFO overrun	The number of times the serial interface card could not send data fast enough to the system packet buffer memory to receive packets from the network.
L & F bits not set	On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the L & F bits not set counter is incremented, and the buffer is cleared for reuse. **Note:** It is unlikely that the L & F bits not set counter will be affected by traffic.

Output frame counters:

DMA/FIFO underrun errors	Number of times the serial interface card could not retrieve data fast enough from the system packet buffer memory to transmit packets onto the network.
Output aborts sent	Number of transmissions that were aborted as requested by upper-level software.

Exit

Use the **exit** command to return to the GWCON (+) prompt.

Syntax: exit

Example: `exit`

V.34 and the GWCON Commands

While V.34 has its own console process for monitoring purposes, the router also displays configuration information and complete statistics for devices and circuits when you use the interface, statistics, and error commands from the GWCON environment. You can also use the GWCON **test** command to test DCEs and circuits.

Note: Issuing the **test** command to the V.34 serial interface causes the current call to be dropped and re-dialed.

For more information on the GWCON command, see Chapter 6, “The GWCON (Monitoring) Process and Commands” on page 6-1.

Statistics for V.34 Interfaces and Dial Circuits

Use the **interface** command at the GWCON (+) prompt to display statistics for V.34 serial line interfaces and dial circuits.

To display the following statistics for a V.34 serial line interface, use the **interface** command followed by the *interface number* of the V.34 serial line interface.

Example: `interface 1`

```
                                    Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
1  1   V.34/0    80000000   44          1          0            0

V.34 MAC/data-link on SCC Serial Line interface

Adapter cable:       RS-232 DTE       RISC Microcode Revision: 1

   V.24 circuit: 105 106 107 108 109 125
   Nicknames:    RTS CTS DSR DTR DCD R1  LL
   RS-232:       CA  CB  CC  CD  CF  CE
   State:        OFF OFF OFF OFF OFF OFF OFF

Line Speed:               115.200 Kbps
Last port reset:          1 hour, 28 minutes, 25 seconds ago

Input frame errors:
   CRC error                    0   alignment (byte length)   0
   missed frame                 0   too long (>  2182 bytes)  0
   aborted frame                0   DMA/FIFO overrun          0

   Output frame counters:
    DMA/FIFO underrun errors    0   Output aborts sent        0
```

To display the following statistics for a dial circuit, use the **interface** command followed by the *interface number* of the dial circuit.

Example:

```
interface 3

                                    Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
3  2   PPP/1        81640   5C          0          5            0

  Point to Point MAC/data-link on V.34 Dial Circuit interface
```

The following table describes the output for both serial line interfaces and dial circuits.

Nt	Serial line interface number or dial circuit interface number.
Nt′	If “Nt” is a dial circuit, this is the interface number of the V.34 serial line interface to which the dial circuit is mapped.
Interface	Interface type and its instance number.
CSR	Command and status register addresses of base network.
Vec	Interrupt vector address.
Self-Test Passed	Number of self-tests that succeeded.
Self-Test Failed	Number of self-tests that failed.

Maintenance: Failed	Number of maintenance failures.
Adapter cable:	Type of adapter cable that is being used.
V.24 circuit:	Circuit numbers as identified by V.24 specifications.
Nicknames	Common names for the circuits.
RS-232	EIA 232 (also known as RS-232) names for the circuits.
State	Current state of the circuits (ON or OFF).
Line speed	The transmit clock speed (approximate).
Last port reset	Length of time since the port was reset.
Input frame errors:	
CRC error	Number of packets received that contained checksum errors and as a result were discarded.
Alignment (byte length)	Number of packets received that were not an even multiple of 8 bits in length and as a result were discarded.
Missed Frame	When a frame arrives at the device and there is no buffer available, the hardware drops the frame and increments the missed frame counter.
too long (> nnnn bytes)	Number of packets received that were greater than the configured frame size and as a result were discarded.
DMA/FIFO overrun	The number of times the serial interface card could not send data fast enough to the system packet buffer memory to receive packets from the network.
L & F bits not set	On serial interfaces, the hardware sets input-descriptor information for arriving frames. If the buffer can accept the complete frame upon arrival, the hardware sets both the last and first bits of the frame, indicating that the buffer accepted the complete frame. If either of the bits is not set, the packet is dropped, the L & F bits not set counter is incremented, and the buffer is cleared for reuse. **Note:** It is unlikely that the L & F bits not set counter will be affected by traffic.
aborted frame	Number of packets received that were aborted by the sender or a line error.
Output frame counters:	
DMA/FIFO underrun errors	Number of times the serial interface card could not retrieve data fast enough from the system packet buffer memory to transmit packets onto the network.
Output aborts sent	Number of transmissions that were aborted as requested by upper-level software.

Chapter 47. Using and Configuring the ISDN Interface

This chapter describes the Integrated Services Digital Network (ISDN) interface on the IBM 2210. It includes the following sections:

ISDN Overview

The ISDN interface software allows you to interconnect routers over ISDN. You can set up the interface to act as a dedicated link or to initiate and accept switched-circuit connections, either on demand, automatically from restart, or on command by the operator.

ISDN Adapters and Interfaces

There are two distinct ISDN interfaces, Basic Rate Interface (BRI) and Primary Rate Interface (PRI). Models 127, 128, 1S4, and 1S8 have an S/T interface integrated on the planar. Models 1U4 and 1U8 have a U interface BRI integrated on the planar, therefore you do not need an NT1 unit to operate in North America.

The following ISDN adapters are available for the 14T, 24T, 24E, and 24M models:

- 1-Port S/T ISDN-BRI
- 4-Port S/T ISDN-BRI
- 4-Port U ISDN-BRI
- 1-Port E1 120-ohm ISDN-PRI
- 1-Port T1/J1 ISDN-PRI

The PRI adapters have an integrated CSU/DSU, so an external CSU/DSU is not required.

The interfaces are:

- Basic Rate Interface (BRI)

  The Basic Rate Interface provides two 64-Kbps (Kilobits per second) bearer (B) channels and one 16-Kbps data (D) channel. The B channels are used as HDLC frame delimited 64-Kbps pipes. The D channel is used to set up calls.

- Primary Rate Interface (PRI)

  The Primary Rate Interface provides functions that are similar to those provided by the Basic Rate Interface. However, there are some important differences:

  – The PRI adapter does not support multipoint. The BRI adapter does.

  – The PRI adapter provides T1/J1 and E1 support.

    - T1/J1 supports twenty-three 64-Kbps B channels and one 64-Kbps D channel.

- E1 supports thirty 64-Kbps B channels and one 64-Kbps D channel.

The ISDN interface establishes connections with a peer router over an ISDN connection. The interface accepts or initiates connections on command from dial circuits. Once the connection is established, the ISDN interface transparently passes data to and from the dial circuit.

Dial Circuits

There are three types of dial circuits:

- Static circuits (or link)
- Switched circuits that dial on demand and hang up after a specified idle time
- WAN restoral circuits that are used only when an assigned primary leased line fails

When bridging over a dial on demand interface it is recommended that you disable spanning tree for that interface and create MAC filters to filter out all undesired traffic. (The MAC filters would drop all frames that are not destined specific MAC addresses.) This keeps the dial circuit from staying connected due to unwanted traffic.

Note: You don't need to add any MAC filters when running BAN traffic on a FR dial-on-demand interface. The BAN software always performs filtering such that the only bridging traffic that will keep a dial-on-demand circuit from hanging up is traffic whose destination MAC address matches the BAN DLCI MAC address.

Add a dial circuit for each potential destination. You can map multiple dial circuits to one ISDN interface. Each dial circuit is a normal serial line network, running Point-to-Point Protocol (PPP) or Frame Relay. These protocols are configured to operate over the dial circuits.

Note: You can assign a destination name to a **connection list** and assign a destination number to each line in the list. When that destination name is called, the numbers in the list are tried one by one until a connection is made or the list is exhausted.

Routable protocols and bridging and routing features cannot communicate directly with an ISDN interface. You need to configure these protocols to run on the dial circuits. This implementation supports the following protocols and features for ISDN dial circuits:

- APPN
- Banyan VINES
- DECnet
- DLSw
- IP
- IPX
- AppleTalk 2
- Bridging (SRB, STP, SR-TB, and SRT)
- Bandwidth reservation
- WAN restoral

Addressing

To place a telephone call, you need to specify the telephone number of the destination. To identify yourself to the switch, you need to specify your own telephone number. For ISDN, telephone numbers are called network dial addresses and, for convenience, they are given names called network address names that represent the telephone number.

When you set up an ISDN interface, you add addresses for each potential destination as well as for your own telephone number, which is called the local network address. When you configure a dial circuit, the local network address is obtained from the physical interface configuration and you set a destination addresses for the circuit.

Circuit Contention

An ISDN BRI T1/J1 can only have 23 active calls at a time and an ISDN PRI E1 can have 30 active calls at one time. Normally, an ISDN BRI can have 2 active calls, except on the 1S4/1S8/1U4/1U8 models when the WAN is also active. In that case, only one call can be active. If the maximum number of dial circuits are active on the ISDN interface, other dial circuits configured for the same interface and the same priority cannot use it. The router drops packets sent by protocols on dial circuits that cannot connect to the ISDN destination.

See “Set” on page 49-4 for more information about priority.

Cost Control Over Demand Circuits

Dial-on-demand circuits always appear to be in the Up state to the protocols. Most protocols send out periodic routing information that could cause the router to dial out each time the routing information is sent over dial-on-demand circuits. To limit periodic routing updates, configure IP and OSI to use only static routes and disable the routing protocols (RIP, OSPF) over the dial circuits. If you are using IPX, configure static routes and services and disable the routing protocols (RIP, SAP) over the dial circuits. Another option is to configure low-frequency RIP and SAP update intervals, although this does not prevent RIP and SAP from broadcasting routing information changes as they occur. You should also enable IPX Keepalive filtering, which prevents keepalive and serialization packets from continually activating the dial-on-demand link.

Call Verification

This ISDN implementation uses a proprietary caller-ID protocol to match incoming calls to dial circuits. The ID protocol uses the inbound and line ID name in the dial circuit configuration to match the dial circuit placing the call to the dial circuit that is receiving the call. The caller-ID protocol is a brief identification protocol initiated by the caller and answered by the dial circuit receiving the call. If the caller does not provide the caller-ID message, the call may be rejected. The line ID exchanges occur on the B channel.

When connecting to routes that do not support logical ids (LIDS), you can suppress the lid exchange using the config option under the individual dial circuit config

```
set lid_used
```

.

. On the incoming side, if this variable is set, the call is transferred to the first dial circuit configured for any inbound or with the caller's phone number in the inbound destination field.

ISDN Cause Codes

This ISDN implementation specifies a cause code that will stop the router from attempting to establish a connection through an ISDN interface. If the application retries, the router again attempts to establish a connection through this interface and will succeed if the original problem has been corrected. If during the retry the router encounters the same cause code, the application will not attempt further connection processing through this interface.

Cause code interpretations:

1. If cause0 is not "0x5" ignore the cause code.
2. If cause0 is "0x5" look at cause1. If the high-order (most significant) bit of cause1 is `ON`, set it to `OFF`.
3. Convert the result to decimal and look up the meaning in the following table, which is taken from *ITU-T Recommendation Q.850*.

Table 47-1 (Page 1 of 2). ISDN Q.931 Cause Codes

Code	Cause
1	Unallocated (unassigned number)
2	No route to specified transit network
3	No route to destination
6	Channel unacceptable
7	Call awarded and is being delivered in an established channel
16	Normal call clearing
17	User busy
18	No user responding
19	No answer from user (user alerted)
21	Call rejected
22	Number changed
26	Non-selected user clearing
27	Destination out of order
28	Invalid number format (address incomplete)
29	Facility rejected
30	Response to STATUS ENQUIRY
31	Normal, unspecified
34	No circuit/channel available
38	Network out of order
41	Temporary Failure
42	Switching equipment congestion
43	Access information discarded
44	Requested circuit/channel not available

Table 47-1 (Page 2 of 2). ISDN Q.931 Cause Codes

Code	Cause
47	Resource unavailable, unspecified
49	Quality of Service not available
50	Requested facility not subscribed
57	Bearer capability not authorized
58	Bearer capability not presently available
63	Service or option not available, unspecified
65	Bearer capability not implemented
66	Channel type not implemented
69	Requested facility not implemented
70	Only restricted digital information bearer capability is available
79	Service or option not implemented, unspecified
81	Invalid call reference value
82	Identified channel does not exist
83	A suspended call exists, but this call identity does not
84	Call identity in use
85	No call suspended
86	Call having the requested call identity has been cleared
88	Incompatible destination
91	Invalid transit network selection
95	Invalid message, unspecified
96	Mandatory information element is missing
97	Message type nonexistent or not implemented
98	Message not compatible with call state or message type nonexistent or not implemented
99	Information element nonexistent or not implemented
100	Invalid information element contents
101	Message not compatible with call state
102	Recovery on timer expiry
111	Protocol error, unspecified
127	Interworking, unspecified

Sample ISDN Configurations

The following topics show several typical ISDN configurations.

ISDN Connection with Four Routers

Figure 47-1 shows a sample ISDN configuration where each router has dedicated ISDN connections to two of its peers. Each link is bidirectional, forming a pair of counter-rotating rings, so that if any one link or router fails, none of the other routers is isolated.

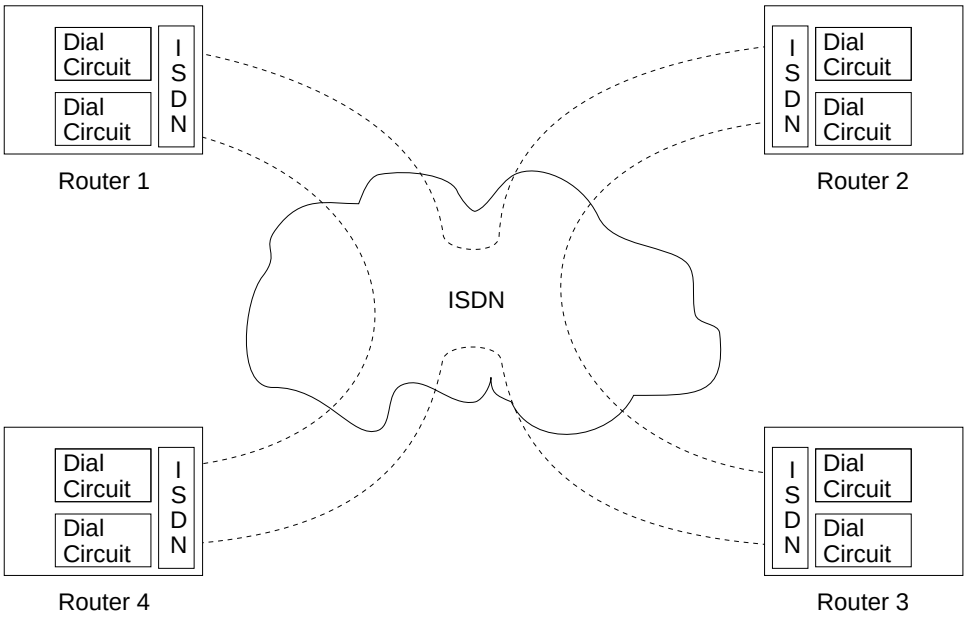


Figure 47-1. Sample ISDN Connection with Four Routers

Point-to-Point Configurations

In the ISDN configurations in Figure 47-2 and Figure 47-3, both routers are in a point-to-point configuration, where there is one router on the ISDN line. In this case, the routers can use the B channels.

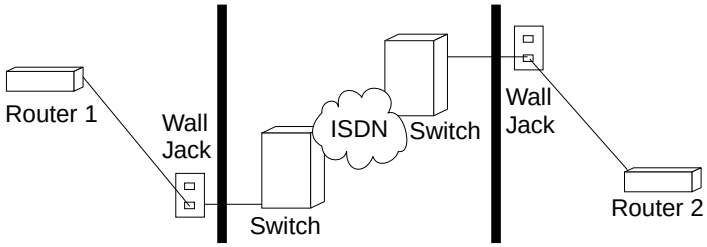


Figure 47-2. ISDN Point-to-Point Configuration (except BRI S/T)

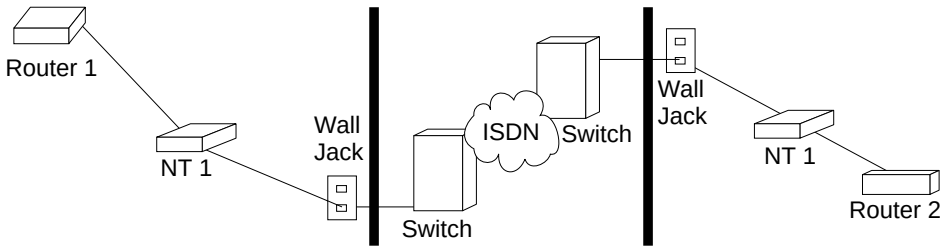


Figure 47-3. ISDN Point-to-Point Configuration (BRI S/T only)

Multipoint Configurations

The Basic Rate Interface (BRI) supports multipoint connections. The Primary Rate Interface (PRI) does not.

In Figure 47-4, Router 1 is sharing the ISDN line with another device in a multipoint configuration. Router 1 and the other ISDN device each can use one of the B channels.

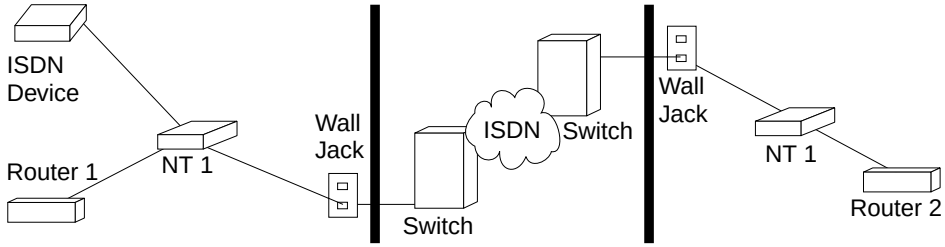


Figure 47-4. ISDN Multipoint Configuration

Frame Relay over ISDN Configuration

Figure 47-5 shows how you can connect to a Frame Relay network through an ISDN network. In this configuration, you set the data link on your dial circuits to Frame Relay.

Note: Dial circuits default to point-to-point (PPP) protocol. To change the protocol to Frame Relay, enter **set data-link fr** at the `Config>` prompt. A connection will only be usable if the data link on both ends matches (for example, either FR to FR, or PPP to PPP).

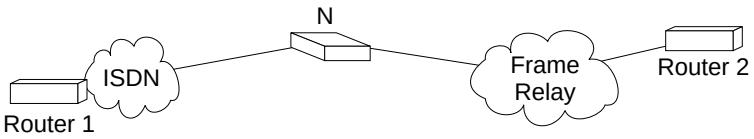


Figure 47-5. Frame Relay over ISDN Configuration

WAN Restoral Configuration

Figure 47-6 on page 47-8 shows how you can use an ISDN connection to back up a failed dedicated WAN link (WAN restoral). In this example, Router A normally uses the WAN link to communicate with Router B. If that connection fails, the ISDN dial-up link reconnects the two routers. When the WAN link recovers, the secondary link automatically disconnects. For more information on how to configure the router for WAN restoral, see Chapter 14, “Configuring WAN Restoral” on page 14-1.

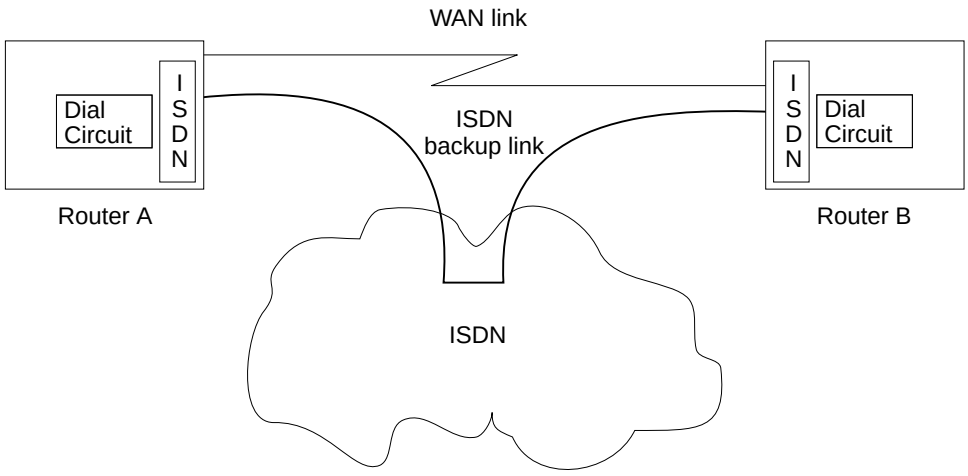


Figure 47-6. Using ISDN for WAN Restoral

For WAN Restoral, only dial circuits configured for PPP can be used as the secondary link. For WAN Reroute, either a PPP dial circuit or a FR dial circuit can be used as the alternate link.

Requirements and Restrictions for ISDN Interfaces

Router

The ISDN software requires the following models of the IBM 2210:

- 127
- 128
- 14T
- 24E
- 24T
- 24M
- 1S4
- 1S8
- 1U4
- 1U8

Switches Supported

The ISDN Basic Rate Interface (BRI) supports the following switches:

- AT&T 5ESS (United States)
- DMS100 (United States)
- USNI1 (United States National ISDN1)
- USNI2 (United States National ISDN2)
- NET 3 (European ETSI)
- INS-Net 64 (Japan)
- VN3 (France Telecom)
- AUS TS 013 (Australia)
- I.430 (See “ISDN I.430 and I.431 Switch Variants” on page 47-14.)

The ISDN Primary Rate Interface (PRI) supports the following switches:

Switch names	Valid command
AT&T 5ESS (United States)	5ESS
AT&T 4ESS	4ESS
Australia (AUSTEL)	AUSPRI
INS-Net 1500 (Japan, NTT)	INSPRI
National ISDN 2	USNI2
NET 5 (Euro-ISDN, ETSI)	NET5
Northern Telecom 250 (DMS250)	DMS250
Native I.431	I431 (See “ISDN I.430 and I.431 Switch Variants” on page 47-14.)

ISDN Interface Restrictions

- You cannot boot or dump the router over an ISDN interface.
- You cannot use the D channel for data traffic. The D channel is used only for setting up and taking down B channel connections.
- Optional ISDN network provider-supplied X.25 connectivity is not supported on the D channel.
- The option to multiplex two B channels to create a single, 128Kbps channel is supported natively only for I.430. You can use the Multilink Protocol (MP – see Chapter 35, “Using and Configuring the Multilink PPP Protocol” on page 35-1) to multiplex any ISDN channel.

Dial Circuit Configuration Requirements

You need to consider the following when you configure PPP or Frame Relay with ISDN:

- The ISDN interface will not enforce transmit delay counters that you set in the PPP configurations.
- Do not enable psuedo-serial-ethernet on the dial circuit.

Before You Begin

Before you configure ISDN, you need the following information:

- Telephone number of the local ISDN port.
- Destination telephone numbers, including any telephone extensions.
- Type of switch to which the ISDN interface is connected. See “Switches Supported” on page 47-8 for the list of switches.

Note: TEI, SPID, and multipoint are not required for PRI.

Configuration Procedures

This section describes how to configure your ISDN interface and its associated dial circuits. Specifically, the tasks you need to perform are:

1. Adding ISDN addresses
2. Configuring ISDN parameters
3. Configuring the ISDN Interface (PRI only)
4. Adding dial circuits
5. Configuring dial circuits

Note: You must restart the router for configuration changes to take effect.

Adding ISDN Addresses

You need to add an ISDN address for each ISDN interface as well as for each destination. The ISDN address includes:

- *Address Name*. The address name is a description of the address. You can use any string of up to 23 printable ASCII characters.
- *Network Dial Address*. Telephone number of the local or destination port. You can enter up to 25 numbers as well as 6 characters, including punctuation. The router uses only the numbers.
- *Network Subdial Address*. Optional. This is an additional part of telephone number, such as an extension, that is interpreted once the interface connects to a PBX. You can enter up to 20 numbers, as well as 11 additional spaces and punctuation. The router uses only the numbers.

To add an ISDN address, enter the **add isdn-address** command at the `Config>` prompt. For example:

```
Config>add isdn-address
Assign address name [23] chars []? baltimore
Assign network dial address [1-15 digits] []? 1-555-0983
Assign network subdial address [1-20 digits] []? 23
```

To see a list of your ISDN addresses, enter **list isdn-address** at the `Config>` prompt.

To delete an ISDN address from your list, enter the **delete isdn-address** command at the `Config>` prompt.

Configuring ISDN Parameters

Access the `ISDN Config>` prompt. To access the `ISDN Config>` prompt, enter the **network** command followed by the interface number of the ISDN interface at the `Config>` prompt. For example:

```
Config>network 3
ISDN user configuration
ISDN Config>
```

You can use the **list devices** command at the `Config>` prompt to display a list of interface numbers configured on the router. See “ISDN Configuration Commands” on page 47-15 for more information about configuration commands.

1. Specify the type of switch to which this ISDN interface is connected.

   Use the **set switch-variant** command to specify the type of switch to which this ISDN interface is connected. See “Switches Supported” on page 47-8 for the list of switches. For example:

   ```
   ISDN Config>set switch net5
   ```

2. Specify the network address name of the local port.

   Use the **set local-address-name** command to specify the network address name of the local port. You must use one of the address names you defined using the **add isdn-address** command. For example:

   ```
   ISDN Config>: set local-address-name
   Assign local address name []? baltimore
   ```

3. Set the directory number of the local port.

Enter the network dial address (telephone number) of the ISDN address that you entered using **set local-address-name** in 2.

```
ISDN Config>set dn0
Enter DN0 (Directory-Number-0) [ ]?15550983
```

4. For BRI only, set the ISDN interface to either point-to-point (pp) or multipoint (mp).

   Point-to-point is one ISDN device on an ISDN line. Multipoint is two or more ISDN devices sharing an ISDN line. With some switch variants, you must configure the line as multipoint regardless of how many devices are on it. Check with your ISDN service provider.

   ```
   ISDN Config>set multi-point-selection
   Multipoint Selection [MP]? pp
   ```

   Note: You can only configure point-to-point for a PRI.

5. For BRI only, if you are connected to a U. S. switch variant, your service provider may require a Service Profile ID (SPID).

   The SPID is a number up to 20 digits long that uniquely identifies the ISDN device. Your ISDN service provider assigns SPIDs.

   ```
   ISDN Config>set spid
   Enter BChannel Number [1]? 1
   Enter Service Profile ID (SPID) []? 123
   ```

6. For BRI only, set the TEI (Terminal Endpoint Identifier) to match the signalling TEI number of your ISDN switch.

   Check with your service provider to find out what TEI signalling the switch supports. The default TEI is auto. If the switch to which your ISDN interface is connected does not support automatic TEI signalling, you must set the TEI to a value from 0 to 63.

   If you are connected to a 5ESS or USNI1 switch, you must set the TEI for each B-channel. The **set tei** command prompts you for a B-channel number.

   ```
   ISDN Config>set tei
   TEI [AUTO]? 10
   ```

   Note: TEI for a PRI is always 0.

7. To set the frame size, use the **set framesize** command. For example:

   ```
   ISDN Config>set framesize
   Framesize in bytes (1024/2048/4096/8192) [1024]? 2048
   ```

   Note: If you choose a frame size of 1024, PPP will not work over the ISDN dial circuit, since the minimum frame size for PPP is 1500.

   For more information about setting the ISDN framesize, see “Set” on page 47-17.

Optional ISDN Parameters

This section describes optional ISDN parameters you can set. For a complete description of these commands see “ISDN Configuration Commands” on page 47-15.

- For all ISDN switches except INS64, you can limit the number of calls to an address. USE the **set retries-call-address** command to set the number of calls to a non-responding destination. Use the **set timeout-call-address** command to set the time period to wait before trying the call again. If you set retries to 0, all circuits will try to come up simultaneously if your ISDN telephone service

provides accounting information, you can use the **add accounting entry** command to keep track of telephone charges.

When you have finished configuring the ISDN interface, you can use the **list** command to display your configuration.

Adding Dial Circuits

Dial circuits are mapped to ISDN interfaces. You can map multiple dial circuits to one ISDN interface.

To add a dial circuit, enter the **add device dial-circuit** command at the `Config>` prompt. The software assigns an interface number to each circuit. You will use this number to configure the dial circuit. For example:

```
Config>add device dial-circuit
Adding device as interface 6
```

The number of dial circuits that can be configured depends on the total number of parameters to be configured and the size of the resulting configuration file up to a maximum of 65 dial circuits per physical interface .

Note: Dial circuits default to point-to-point (PPP) protocol. To change the dial circuit protocol to Frame Relay, enter the **set data-link fr** command at the `Config>` prompt. Other data-link types (X.25, SDLC, and SRLY) are not supported.

Configuring Dial Circuits

This section describes how to configure a dial circuit.

1. Display the `Circuit Config>` prompt by entering the **network** command followed by the interface number of the dial circuit. You can enter the **list devices** command at the `Config>` prompt to display a list of the interface numbers configured on the router. For example:

   ```
   Config>network 6
   Circuit configuration
   Circuit Config>
   ```

2. Map the dial circuit to an ISDN interface. Use the **set net** command. The Base net is the ISDN interface number. For example:

   ```
   Circuit Config>set net
   Base net for this circuit [0]? 3
   ```

3. Specify the address name of the remote router to which the dial circuit will connect. You must use one of the names you defined using the **add isdn-address** command. For example:

   ```
   Circuit Config>set destination
   Assign destination address name []? baltimore
   ```

4. Configure the dial circuit to initiate outbound calls only, accept inbound calls only, or to both initiate and accept calls.

   Use the **set calls** command. To avoid a conflict if both ends of the link attempt to establish a call at the same time, configure the dial circuit at one end of the link to accept inbound calls only, and configure the dial circuit at the other end of the link to initiate outbound calls only. For example:

   ```
   Circuit Config>set calls outbound
   Circuit Config>set calls inbound
   Circuit Config>set calls both
   ```

Note: For WAN-Restoral operations or another dial-on-demand application, you should set up the circuit for either inbound or outbound calls.

5. Specify the timeout period for the circuit.

   Use the **set idle** command. If there is no traffic over the circuit for this specified time period, the dial circuit hangs up. To configure the circuit as a dedicated circuit, set the idle timer to zero. To configure the circuit to dial on demand, set the idle timer to a value other than zero. The range is 0 to 65535 and the default is 60 seconds. For example:

   ```
   Circuit Config>set idle
   Idle timer (seconds, 0 means always active) [0]? 0
   ```

6. Optionally, you can provide a name for a dial circuit by specifying a lid_out_addr.

   When more than one circuit is configured between two routers (parallel circuits), there must be a way to know which dial circuit connects them. For this purpose, a lid_out_addr is sent from the router at one end (the caller). The receiving router must have an inbound destination address that matches the lid_out_address on the sending router in order for the dial circuits to connect. The lid_out_addr must be an address name that has been previously added using “ADD ISDN-ADDRESS” at the **config**> prompt.

   ```
   Circuit Config>set lid_out_addr  router2
   ```

7. Optionally, you can set the relative priority of dial circuits.

   The priority field allows a circuit to preempt another when no channels are available. If a call request is made and all the channels are in use, then the priority of the requesting dial circuit is checked against all the active dial circuits. If there is one whose priority is lower than this, then that circuit is disconnected and a call is made for the higher priority dial circuit.

   Note: Only outbound dial-on-demand circuits will be brought down.

   See “Set” on page 49-4 for more information about priority.

   ```
   Circuit Config>set priority 1
   ```

8. Optionally, you can delay the time between when a call is established and the initial packet is sent. Use the **set selftest-delay** command. Some ISDN switches start to send data before receiving a signal indicating the complete establishment of the circuit at the destination. Setting a selftest delay can prevent initial packets from being dropped. For example:

   ```
   Circuit Config>set selftest-delay
   Selftest delay(milli-seconds,0 means no delay)[150]?200
   ```

9. Set the inbound address name.

   Use the **set inbound** command. This command is for inbound circuits only. For example:

   ```
   Circuit Config> set inbound
   Assign destination inbound address name [ ]? newyork
   ```

   The inbound address name must match one of the names you defined using the **add isdn-address** command.

10. Optionally, you can enter the configuration process for the data-link layer protocol that is running on the dial circuit (PPP or Frame Relay).

    Use the **encapsulator** command. For example:

    ```
    Circuit Config> encapsulator
    ```

ISDN I.430 and I.431 Switch Variants

To use the Native I.430 mode that is supported in Japan and is known as D64S in Germany, you must code the ISDN switch variant as I.430. This treats the ISDN interface like a leased line. There is no D-channel signalling traffic in this mode.

The I.431 switch variant should be configured when running Primary Rate ISDN over leased lines.

Native I.430 Support

When configuring for Native I.430 support,only one dial circuit should be used. It should be attached to the base net. The speed can be configured to 64–Kbps or 128–Kbps using the `set bandwidth` command. On models 1S4, 1S8, 1U4, and 1U8, if WAN and ISDN are both active, this is restricted to 64Kbps only. See “Set” on page 47-17 to configure the bandwidth command.

Example: Base ISDN Net

```
Config>n 6
ISDN Config>set switch i430
ISDN Config>list all

                                    ISDN Configuration

Maximum frame size in bytes      = 2048
Switch Variant                   = I430 BRI
PS1 detect                       = Enabled
```

Example: Dial Circuit

```
Config>n 7 ------------------ DIAL CIRCUIT (CAN ONLY BE ONE FOR I430/I431)
Circuit config:   7>
Circuit config:   7>set net 6

Circuit config:   7>list all

Base net                       = 6
I430 BRI Bandwidth             = 128 kbs
```

Native I.431 Support

When configuring for Native I.431 support, only one dial circuit should be used. It should be attached to the base net. The I.431 only runs on the ISDN PRI T1 adapter. The speed is fixed at 1.5 Mbps.

Example: Base ISDN net

```
Config>n 5
ISDN Config>set sw i431
ISDN Config>list all
                   ISDN Configuration

Maximum frame size in bytes      = 2048
Switch Variant                   = I431 PRI
```

Example: Dial Circuit

```
Config>n 6
Circuit config:   6>set net 5
Circuit config:   6>list all


Base net                       = 5
```

ISDN Configuration Commands

Table 47-2 summarizes the ISDN configuration commands, and the following sections explain the commands. Enter these commands at the `ISDN Config>` prompt.

Table 47-2. ISDN Configuration Command Summary

Command	Function
? (Help)	Lists the configuration commands or lists the options associated with that command.
Add	Adds accounting entries to the ISDN configuration.
Disable	Valid only for BRI. Disables Power Source 1 detection.
Enable	Valid only for BRI. Enables Power Source 1 detection.
List	Displays the ISDN configuration.
Remove	Removes accounting entries from the ISDN configuration.
Set	Sets the frame size, local address, no-answer timeouts, number of retries after no answer, type of ISDN switch, directory numbers, TEI and bandwidth.
Cause Codes	Stops further processing attempts to establish a connection through an interface.
Exit	Exits the ISDN configuration process and returns to the `Config>` prompt.

? (Help)

The **?** (help) command lists available commands. You can also enter ? after a command to list its options.

Syntax: ?

Example: **`set ?`**

```
FRAMESIZE
LOCAL-ADDRESS-NAME
RETRIES-CALL-ADDRESS
TIMEOUT-CALL-ADDRESS
SWITCH-VARIANT
DN0 (Directory Number 0)
INTERFACE
```

Add

The **add** command lets you add accounting entries to your ISDN configuration.

Syntax: add accounting-entry ...

accounting entry *name*

If your ISDN telephone service provides accounting information, you can use accounting entries to track telephone charges accrued for specified network addresses. You can add up to eight entries for each ISDN interface. The accounting entry name must match one of the ISDN addresses you added using the **add isdn-address** command at the `Config>` prompt.

Example: **`add accounting-entry`**

```
Assign accounting entry name []? baltimore
```

To display accrued telephone charges, enter the accounting command at the ISDN console prompt.

Disable

The **disable** command disables Power Source 1 detection. If your switch does not supply Power Source 1, you should disable PS1.

Note: This command is valid only for BRI.

Syntax: disable ps1

Example: `disable ps1`

Note: On the U interface ISDN BRIs, there is no ps1 detect circuitry and the value of this field is ignored.

Enable

The **enable** command enables Power Source 1 detection. If your ISDN switch supplies Power Source 1 (PS1), you should enable PS1 on the interface. This causes the interface to detect when the switch shuts down and to clear all information about the last call before it reestablishes the connection.For Euro-NET3 switches supporting restricted power mode, PS1 must be enabled.

Do not enable PS1 if your switch does not supply Power Source 1.

Note: This command is valid only for BRI.

Syntax: enable ps1

Example: `enable ps1`

Note: On the U interface ISDN BRIs, there is no ps1 detect circuitry and the value of this field is ignored.

List

The **list** command displays the current ISDN configuration.

Syntax: list

Example: `list`

```
                                ISDN Configuration
  Local Network Address Name    = line-1-local
  Local Network Address         = 1-508-555-1234
  Local Network Subaddress      = 21

  Maximum frame size in bytes   = 1024
  Outbound call address Timeout =    0  Retries =  0
  Switch-Variant-Model          = ETSI NET3
  Multipoint Selection          = Point-to-Point
  DN0 (Directory Number 0)      = 1-508-555-1234
  DN1 (Directory Number 1)      = 1-508-555-3456
  Service Profile ID (B1)       = 123
  Service Profile ID (B2)       = 456
  TEI for B-Channel 1           = Automatic
  TEI for B-Channel 2           = Automatic
  PS1 detect                    = Disabled

  Accounting information kept for the following network destinations:

  Address assigned name      Network Address  Network
  Subaddress
  ---------------------      ---------------  -----------
  remote-ny1                     100              100
  remote-ny2                     200              200
```

Example: `list`

```
                                  ISDN Configuration

Local Network Address Name       = local2210
Local Network Address:Subaddress = 2542210:

Maximum frame size in bytes      = 2048
Outbound call address Timeout    =    0  Retries =   0
Switch Variant                   = NT DMS-250
DN0 (Directory Number 0)         = 2542210
No circuit address accounting information being kept

T1/J1 Interface Parameters:

LBO               = 00.0 dB
Code              = B8ZS
ZBTSI             = Disabled
ESF-Data-Link     = ANSI-IDLE
```

Remove

The **remove** command lets you remove accounting entries you set using the **add accounting-entry** command.

Syntax: remove accounting-entry...

Example: `remove accounting-entry`

```
Remove accounting entry name []? baltimore
```

Set

The **set** command configures frame size, addresses, and timeouts. It also specifies the switch-variant and TEI number. For PRI, the terminal endpoint identifier (TEI) is always zero (0).

Syntax: set bandwidth[1]...
framesize

[1] I.430 only

interface
local-address-name...
multipoint-selection[2]...
retries-call-address...
service-profile-id[2]...
timeout-call-address[2]...
switch-variant...
dn0...
dn1[3]...
tei[2]...

bandwidth

Sets the speed of the I.430 dial circuit. When the speed is set at 64-Kbps, only the B1 channel is used. When the speed is set at 128-Kbps, The B1 and B2 Channel are combined to form a single pipe.

Example: **set bandwidth**

```
Circuit config: 10>set 64 or 128[64]? 128
```

framesize *1024* or *2048* or *4096* or *8192*

Sets the size of the network layer portion of frames transmitted and received on the ISDN interface. Data link and MAC layer headers are not included. You must set the ISDN frame size so that it is greater than or equal to the frame size configured for the dial circuits using the ISDN interface.

For PPP dial circuit interfaces, you can change the PPP MRU using the **set lcp options** command. The ISDN frame size must include enough bytes for the PPP MRU and the PPP header.

Note: If you choose a frame size of 1024, PPP will not work over the ISDN dial circuit, since the minimum frame size for PPP is 1500.

For FR dial circuit interfaces, you can change the frame size using the **set framesize** command. The ISDN frame size must be greater than or equal to the FR frame size.

If a dial circuit's frame size is greater then the ISDN frame size, then the dial circuit's frame size is decreased at router initialization.

Example: **set framesize**

```
Framesize in bytes (1024/2048/4096/8192) [1024]? 2048
```

interface

For PRI only. Sets the following interface parameter values for T1 and E1 lines.

For T1 PRI:

lbo — The attenuation of the signal transmitted by the router's T1 port. This information is provided by the service provider.

Valid Values:

a= -00.0 dB
b= -07.5 dB

---

2 BRI only

3 PRI only

c= -15.0 dB
d= -22.5 dB

Default Value: a

code — This information is provided by the service provider.

Valid Values: B8ZS or AMI

Default Values: B8ZS

ZBTSI — Zero Byte Time Slot Inversion. This information is provided by the service provider.

Valid Values: Enabled or Disabled

Default Value: Disabled

esf-data-link — The service subscription. This information is provided by the service provider.

Valid Values:

ANSI-T1.403
ANSI-IDLE
AT&T-IDLE

Default Value: ANSI-T1.403

For E1 PRI:

code — This information is provided by the service provider.

Valid Values: HDB3 or AMI

Default Value: HDB3

crc4 — Specifies whether the router's E1 port will transmit crc4 code words and check them in the received frames. This information is provided by the service provider.

Valid Values:Enabled or Disabled

Default Value:Disabled

local-address-name *address name*

This is the network address name of the local ISDN interface. This address name must match one of the names that you defined at the `Config>` prompt using the **add isdn-address** command.

Valid Values: Any valid address

Default Value: None

Example: **`set local-address-name`**

```
Assign local address name []? line-1-local
```

multipoint-selection *mp* or *pp*

For BRI only. Sets the ISDN physical bus to either point-to-point (pp) or multipoint (mp) configuration. Point-to-point is one ISDN device on an ISDN line. Multipoint is two or more ISDN devices sharing an ISDN line.

Some service providers require that you configure the line as multipoint regardless of how many devices are on the line. Check with your ISDN service provider.

Example: `set multipoint-selection`

```
Multipoint Selection [PP]? mp
```

retries-call-address *value*

Some telephone service providers impose restrictions on automatic recalling devices to limit the number of successive calls to an address that is inaccessible or that refuses those calls. **Retries-call-address** specifies the maximum number of calls the router attempts to make at one time . Setting **retries-call-address** to 0 causes the router to bring up all circuits at once.

If you set the switch-variant to INS64,

Valid Values: 0 to 30

Default Value: 23

Example: `set retries-call-address`

```
Outbound call address retries [0]? 2
```

service-profile-id *B-channel# spid#*

For BRI only. Sets the service profile ID (SPID) for each B-channel. SPIDs are used in the United States to uniquely identify a particular ISDN device. This ID is a number up to 20 digits long and is assigned by ISDN service providers. SPIDs are used predominantly in a multipoint bus configuration where multiple ISDN devices share a single ISDN line. Check with your service provider to determine whether or not you are required to use a SPID.

Example: `set spid`

```
Enter B-Channel Number [1]? 1
Enter Service Profile ID (SPID) [123]? 2349
```

timeout-call-address *# of seconds*

After the router reaches the maximum number of **retries-call-address** to a non-responding address, it does not make further calls to that address until this time has expired. The timeout period begins when the router attempts the first call to an address. Setting **timeout-call-address** to 0 causes the router to retry until the call is established.

If you set the switch-variant to INS64, you cannot change **timeout-call-address**. It is fixed at 180.

Valid Values: 0 to 65535 seconds

Default Value: 180 seconds

Example: `set timeout-call-address`

```
Outbound call address Time-out (secs) [0]? 180
```

switch-variant

Specifies the model of the switch to which this ISDN interface is connected. The default for the ISDN Primary Rate interface is DMS250. You can choose switch-variants for the ISDN Basic Rate interface or the ISDN Primary Rate interface from the following lists.

Valid Values Basic Rate Interface (BRI):

- 5ESS (United States)
- DMS100 (United States)
- USNI1 (United States National ISDN1)
- USNI2 (United States National ISDN2)

- NET 3 (European ETSI)
- INS 64 (Japan)
- VN3 (France Telecom)
- AUS TS 013 (Australia)
- Native I.430

Default Value: NET 3

Valid Values ISDN Primary Rate Interface (PRI):

- AT&T 5ESS (United States)
- AT&T 4ESS
- Australia (AUSTEL)
- INS-Pri (Japan, NTT)
- National ISDN 2
- NET 5 (Euro-ISDN, ETSI)
- Northern Telecom 250
- Native I.431

Default Value:DMS250

Example: `set switch-variant`

```
Switch-Variant-Model []? net5
```

dn0 *directory number 0*

DN0 must match the network dial address (telephone number) you configured using the **set local-address-name** command. If DN0 is not configured no check is made and all calls will be accepted. If the switch does not provide the called party number in the incoming setup message, DN0 should not be configured.

Example: **`set dn0`**

```
Enter DN0 (Directory-Number-0) [ ]? 5088981234
```

dn1 *directory number 1*

DN1 is a secondary directory number supported by NET3, VN3 and AUS, switch variants. If DN1 is not configured no check is made and all calls will be accepted. If the switch does not provide the called party number in the incoming setup message, DN1 should not be configured.

tei *auto* or *none* or *value*

For BRI only. This command sets the signalling TEI (terminal endpoint identifier) for the ISDN interface. This setting must match the signalling TEI of your switch. For PRI, the TEI is always set to zero (0). Check with your service provider to find out the correct TEI signal. The default is auto. Change this setting only if your switch does not support automatic TEI signalling. The valid settings for TEI are auto or a value from 0 to 63. If you set the TEI to none, you will disable the ISDN interface.

USNI-1 and 5ESS switches require that you set the TEI for each B-channel. If you set the switch variant to one of those switches, the **set tei** command prompts you for a B-channel number.

Example 1: **`set tei`**

```
TEI  [AUTO]? 60
```

Example 2: **`set tei`**

```
TEI 0 or TEI 1 [1]? 1
TEI [AUTO]?
```

Cause Codes

Use the **Cause Code** command to prevent the router from retrying to establish a connection through the ISDN interface when it receives a “specified” (valid value) response. Enter these commands at the `Cause Config>`prompt.

Table 47-3. ISDN Cause CodesCommand Summary

Command	Function
Add	Adds cause code entries to the ISDN configuration.
List	Displays the cause code lists for the ISDN configuration.
Remove	Removes cause code entries from the ISDN configuration.
Exit	Exits the ISDN cause code configuration process and returns to the `Config>` prompt.

Syntax: ca use

Example: **`cause ?`**

```
ADD
REMOVE
LIST
EXIT
```

Add — Use the **add** command to add a cause code to an ISDN configuration.

Valid Values: Any hexadecimal value between 01 and FF

Default Value: None

Syntax:ca use code *add*

Example: **`add FF`**

Remove — Use the **remove** command to remove a cause code from an ISDN configuration.

Valid Values: Any hexadecimal value between 01 and FF

Default Value: None

Syntax: ca use code *remove*

Example: **`remove FF`**

List — Use the **list** command to show the cause code list of an ISDN configuration.

Syntax: ca use code *list*

Example: **`list`**

Exit — Use the **exit** command return to the *ISDN Config >*prompt.

Syntax: e xit

Example: **`exit`**

Exit

Use the **exit** command to return to the `Config>` prompt.

Syntax: exit

Example: **`exit`**

Chapter 48. Monitoring the ISDN Interface

This chapter describes the ISDN console commands and GWCON commands. It includes the following sections:

- "Accessing the Interface Console Process"
- "ISDN Console Commands"
- "ISDN and the GWCON Commands" on page 48-7

Note: ISDN interfaces also have ELS messages and cause codes that you can use to monitor ISDN-related activity. See *Event Logging System Messages Guide*

Accessing the Interface Console Process

To access the interface console process for ISDN, enter the following command at the GWCON (+) prompt:

```
+ network #
```

Where # is the number of the ISDN interface. You cannot directly access the console process for dial circuits, but you can monitor the dial circuits that are mapped to the ISDN interface.

ISDN Console Commands

The following sections explain the ISDN console commands which allow you to view the accounting entries, calls, circuits, parameters, and statistics of the ISDN interfaces. Enter these commands at the ISDN> prompt.

Table 48-1. ISDN Console Command Summary

Console Command	Function
? (Help)	Lists the ISDN console commands or lists the options associated with specific commands.
Accounting	Displays the telephone charges that have accrued for each network address configured by the **add accounting entries** ISDN configuration command.
Calls	List the number of completed and attempted connections made for each dial circuit mapped to this interface since the last time statistics were reset on the router.
Channels	Lists the statistics for the channels on the ISDN Primary Rate Interface.
Circuits	Shows the status of all data circuits configured on the ISDN interface.
Parameters	Displays the current parameters for the ISDN interface.
Statistics	Displays the current statistics for the ISDN interface.
Exit	Exits the ISDN console process and returns to the GWCON (+) process.

? (Help)

The **? (help)** command lists available commands. You can also enter ? after a command to list its options.

Syntax: ?

Example: **?**

```
ACCOUNTING
CALLS
CIRCUITS
PARAMETERS
STATISTICS
CAUSE_CODE
CHANNELS
EXIT
```

Accounting

If you set up accounting entries using the **add accounting entries** ISDN command, the **accounting** command displays accrued telephone charges for each network address that you added.

Syntax: accounting

Example: **accounting**

```
Address Name    Address         SubAddress          Charge
vl2-31          21              1                    0.0
vl2-33          20              1                    0.0
vl_2-31         021             001                  0.0
vl_2-33         020             001                  0.0
ydc100          100                                  0.0
ydc200          200                                  0.0

All others:                                          0.0
```

Calls

Use the **calls** command to list the number of completed and attempted connections made for each dial circuit mapped to this interface since the last time statistics were reset on the router.

Syntax: calls

Example: **calls**

```
Net Interface  Site Name           In    Out   Rfsd  Blckd
  4     PPP/1  v403                2      0      0      0

Unmapped connection indications:    0
```

Net Number of the dial circuit mapped to this interface.

Interface Type of interface and its instance number.

Site Name Network address name of the dial circuit.

In Inbound connections accepted for this dial circuit.

Out Completed connections initiated by this dial circuit.

Rfsd Connections initiated by this dial circuit that were refused by the network or the remote destination port.

Blckd Connection attempts that the router blocked. The router blocks connection attempts if all available channels are in use, if the max retries are used up and the router is waiting for the timer to count down, or if layer 1 is up, but layer 2 is down.

Unmapped connection indications:
Connection attempts that were refused by the router because there were no enabled dial circuits that were configured to accept the incoming calls.

Channels

The **channels** command lists the statistics for a channel on the ISDN Primary Rate Interface.

Circuits

The **circuits** command shows the status of the dial circuits configured on the ISDN interface that are in the state of “Up” or “Available.”

Syntax: circuits

Example: `circuit`

```
Net Interface  MAC/Data-Link    State     Reason      Duration
4     PPP/1   Point to Point    Up B1     SelfTest    91:24:03
5     PPP/2   Point to Point    Up B2     Inbound     91:24:00
```

Net Number of the dial circuit mapped to this interface

Interface Type of interface and its instance number.

MAC/Data-Link
Type of data-link protocol configured for this dial circuit.

State Current state of the dial circuit:

- *Up* Currently connected.
- *Available* Not currently connected, but available.
- *Disabled* Dial circuit disabled.
- *Down* Failed to connect because of a busy dial circuit or because the link-layer protocol is down.

Reason Reason for the current state:

- *nnn_Data* (Where nnn is the name of a protocol.) The circuit is up because a protocol had data to send.
- *Rmt Disc* Remote Disconnect. The circuit is either down or available because the remote destination disconnected the call.
- *Opr Req* Operator Request. The circuit is available because the last call was disconnected by a console command.
- *Inbound* The circuit is up because the circuit answered an inbound call.
- *Restoral* The circuit is up because of a WAN-Restoral operation.
- *Self Test* The circuit was configured as static (idle time=0) and successfully connected once it was enabled.

Duration Length of time that the circuit has been in the current state.

Parameters

Use the **parameters** command to display the current ISDN configuration.

Syntax: parameters

Example: parameters

```
ISDN Port parameters:

Local Address Name:        vl233
Local Network Address:     20
Local Network Subaddress:
Frame Size:          2048
TEI 0:               Automatic
TEI 1:               Automatic
Switch Variant:      AT&T 5ESS (United States)
Multipoint Selection:    Multipoint
Directory Number 0:      20
Outbound call address Timeout:     180       Retries:     0

Accounting Name           Network Address   Network Subaddress
---------------           ---------------   ------------------
vl215                     22
vl218                     22
vl231                     21
vl233                     20
```

Statistics

Use the **statistics** command to display the current statistics for this ISDN interface.

Syntax: statistics

Example for BRI: statistics

```
Link:   Active    ISDN Firmware:   0.0    Handler State: Running

                      D Channel     B1 Channel     B2 Channel

Total Transmits           32788         230217         164336
Total Receives            32789         164342         208255
Transmit Bytes           196767       22797579        6572177
Receive Bytes            196785        6572411        9517221
Invalid Interrupts            0              0              0

Transmit:     D       B1       B2    Receive:        D       B1       B2

Error         0        0        0    Error           0        5        0
Overflow      0        0        0    Overflow        0        0        0
Underrun      0        0        0    Overrun         0        0        0
Abort         0        0        0    Abort           0        5        0
                                      CRC Error       0        0        0
```

Example for BRI using I.430: statistics

```
Link:   Active    ISDN Firmware:   0.0    Handler State: Running


Total Transmits          32788
Total Receives           32789
Transmit Bytes          196767
Receive Bytes           196785
Invalid Interrupts           0

Transmit:                          Receive:

Error         0                    Error          0
Overflow      0                    Overflow       0
Underrun      0                    Overrun        0
Abort         0                    Abort          0
                                   CRC Error      0
```

This display shows the current state of the link, the firmware revision, and the state of the dial circuit. It also shows statistics on what was transmitted and received on the interface.

Example for PRI with E1: `statistics`

```
  Link:   Active    ISDN Firmware:   1.0    Handler State: Running

Transmit    D  Channel     Receive    D  Channel

Packets           68422     Packets         68419
Bytes            411656     Bytes          413592
Overflow             23     Overflow            3
Underrun              0     Too Long            6
                            Abort               4
                            CRC error           8
                            Misaligned          3

Transmit    B  Channels    Receive    B  Channels

Packets         1499094     Packets       1499228
Bytes          59955660     Bytes        59951780
Overflow              0     Overflow           90
Underrun              0     Too Long          171
                            Abort             139
                            CRC error         232
                            Misaligned         72

E1 Status Register         E1 Error Count Registers

Receive AIS         : Off  CRC6 Errors:        4
Receive RAI         : Off  LCV  Errors:       38
Receive Carrier Loss: Off  FEB  Errors:       11
Receive Loss of Sync: Off  FAS  Errors:       24
```

Example for PRI with T1: `statistics`

```
Transmit    D  Channel     Receive    D  Channel

Packets              0     Packets             0
Bytes                0     Bytes               0
Overflow         68480     Overflow            0
Underrun             0     Too Long            0
                           Abort               0
                           CRC error           0
                           Misaligned          0

Transmit    B  Channels    Receive    B  Channels

Packets              0     Packets             0
Bytes                0     Bytes               0
Overflow             0     Overflow            9
Underrun             0     Too Long            0
                           Abort               0
                           CRC error           0
                           Misaligned          0

T1 Status Register         T1 Error Count Registers

Receive AIS         : Off  LCV  Errors:           0
Receive RAI         : Off  CRC6 Errors:           0
Receive Carrier Loss: Off  Sync Errors:    47937328
Receive Loss of Sync: On

T1 PRM Events                 Local       Remote

CRC Error                         0            0
Controlled Slip                   0            0
Line Code Violation               0            0
Frame Sync Bit Error              0            0
Severely Errored Frame            0            0
Payload Looback Active            0            0
PRMs Processed (1/sec)            0            0
```

Example for PRI with T1 using I.431: `statistics`

```
Transmit                   Receive

Packets              0     Packets             0
Bytes                0     Bytes               0
Overflow         68480     Overflow            0
Underrun             0     Too Long            0
                           Abort               0
                           CRC error           0
                           Misaligned          0

 T1 Status Register         T1 Error Count Registers

Receive AIS         : Off  LCV  Errors:           0
Receive RAI         : Off  CRC6 Errors:           0
Receive Carrier Loss: Off  Sync Errors:    47937328
Receive Loss of Sync: On

T1 PRM Events                 Local       Remote

CRC Error                         0            0
Controlled Slip                   0            0
Line Code Violation               0            0
Frame Sync Bit Error              0            0
Severely Errored Frame            0            0
Payload Looback Active            0            0
PRMs Processed (1/sec)            0            0
```

Exit

Use the **exit** command to return to the GWCON (+) prompt.

Syntax: exit

Example: **exit**

ISDN and the GWCON Commands

While ISDN has its own console process for monitoring purposes, the router also displays configuration information and complete statistics for devices and circuits when you use the **interface, statistics,** and **error** commands from the GWCON environment. You can also use the GWCON **test** command to test DCEs and circuits.

Note: Issuing the **test** command to the ISDN interface causes the current call to be dropped and re-dialed.

Interface — Statistics for ISDN Interfaces and Dial Circuits

Use the **interface** command at the GWCON prompt (+) to display statistics for ISDN interfaces and dial circuits.

To display statistics for a dial circuit, enter the **interface** command followed by the interface number of the dial circuit. For ISDN interfaces, information is displayed on a D and B channel basis. (This is the same information that is displayed by the ISDN **statistics** command.)

Example: **interface 3**

```
                                    Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec      Passed     Failed       Failed
3  3   ISDN/0           0    0          1          0            0

  ISDN Base Net MAC/data-link on ISDN Basic Rate Interface interface
  Link:   Active    ISDN Firmware:   1.0    Handler State: Running

                       D Channel      B Channels

  Total Transmits            591              0
  Total Receives             601              0
  Transmit Bytes            3981              0
  Receive Bytes             4050              0
  Invalid Interrupts           0              0

  Transmit:    D       B Channels     Receive:      D       B Channels
  Error        0               0      Error         0               0
  Overflow     0               0      Overflow      0               0
  Underrun     0               0      Overrun       0               0
  Abort        0               0      Abort         0               0
                                      CRC Error     0               0
```

To display the following statistics for a dial circuit, use the **interface** command followed by the interface number of the dial circuit.

Example: **interface 4**

```
                                        Self-Test  Self-Test  Maintenance
Nt Nt' Interface      CSR  Vec     Passed     Failed       Failed
4  3   PPP/1            0    0          1          2            0

 Point to Point MAC/data-link on ISDN Basic Rate Interface
```

The following list describes the output for both ISDN and dial circuits.

Nt Serial line interface number or dial circuit interface number.

Nt' If *Nt* is a dial circuit, this is the interface number of the ISDN interface to which the dial circuit is mapped.

Interface Interface type and its instance number.

CSR Command and status register addresses of base network.

Vec Interrupt vector address.

Self-Test Passed
Number of self-tests that succeeded.

Self-Test Failed
Number of self-tests that failed.

Maintenance: Failed
Number of maintenance failures.

Configuration — Information on Router Hardware and Software

Enter the **configuration** command at the GWCON (+) prompt to display information about the router hardware and software. It includes a section that displays the interfaces configured on the router along with the state of the interface.

If a dial circuit is configured to dial-on-demand, the state of the dial circuit is always displayed as Up whether or not it is connected. In this case Up means that the dial circuit is either connected or available.

If a dial circuit is configured as a static circuit, the state indicates Up only if the dial circuit is connected. (Refer to “Configuration” on page 6-6 for a sample output from the **configuration** command.

Chapter 49. Configuring Dial Circuits

This chapter describes the dial circuit configuration commands that are relevant to a PPP or Frame Relay dial circuit interface mapped to a V.25bis, V.34, or ISDN interface.

Dial-in and Dial-out interfaces are special types of dial circuit interfaces.

Notes:

1. PPP dial circuit interfaces can use an ISDN, V.25bis, or a V.34 network as the base network interface.
2. FR dial circuit interfaces can use an ISDN or a V.25bis network as the base network interface.
3. Dial-Out circuit interfaces use a V.34 network as the base network interface.
4. Dial-In circuit interfaces can use an ISDN or V.34 network as the base network interface.

For information on how to configure dial circuits for use with:

- ISDN interfaces, see Chapter 47, “Using and Configuring the ISDN Interface” on page 47-1.
- V.25bis interfaces, see Chapter 43, “Using and Configuring the V.25bis Network Interface” on page 43-1.
- V.34 interfaces, see Chapter 45, “Using and Configuring the V.34 Network Interface” on page 45-1.

Dial Circuit Configuration Commands

This section summarizes and explains the dial circuit configuration commands.

Chapter 49, “Configuring Dial Circuits” summarizes the dial circuit configuration commands. Enter the dial circuit configuration commands at the `Circuit Config>` prompt. You must restart the router for configuration changes to take effect.

To access the `Circuit Config>` prompt, enter the **network** command followed by the interface number of the “dial circuit.” (The dial circuit number was assigned when you entered the **add device dial-circuit** command.) You can enter the **list devices** command at the `Config>` prompt to display a list of the dial circuits that you added.

Table 49-1. Dial Circuit Configuration Commands Summary

Command	Function
? (Help)	Lists the configuration commands or lists the options associated with that command.
Delete	Deletes the inbound call settings from the dial circuit configuration.
Encapsulator	Allows you to change the data-link protocol configuration.
List	Displays the dial circuit configuration parameters.
Set	Configures the dial circuit for inbound or outbound calls, maps the dial circuit to a serial line interface, and sets addresses, idle timeout, priority, lid_out address, inbound destination, and self-test delay.
Exit	Exits the dial circuit configuration process and returns to the `Config>` prompt.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a **?** after a specific command name to list its options.

Syntax: ?

Example: **`?`**

```
DELETE
ENCAPSULATOR
LIST
SET
EXIT
```

Example: **`Set ?`**

```
NET
CALLS
DESTINATION
INBOUND DESTINATION
ANY_INBOUND
IDLE
LID_OUT_ADDR
PRIORITY
SELFTEST-DELAY
```

Delete

Use the **delete** command to remove the inbound call settings from the dial circuit configuration.

Syntax: delete inbound destination

inbound destination

Removes both the INBOUND destination and the ANY_INBOUND settings from the dial circuit configuration. This causes the dial circuit to accept calls only from callers that have a phone number that matches the DESTINATION parameter.

Example: **`delete inbound`**

Encapsulator

Use the encapsulator command to enter the configuration process for the link-layer protocol (e.g. PPP, Frame Relay, dial-out) that is running on the dial circuit interface.

Note: The default for a dial circuit interface created via the **add device dial-circuit** command is PPP. If you want to change the link layer type to Frame Relay, use the **set data-link frame-relay** command at the `Config>` prompt.

Syntax: encapsulator

The following example shows that the PPP configuration process is entered when the encapsulator command is used for a PPP dial circuit or dial-in interface.

Example: `encapsulator`

```
Point-to-Point user configuration
PPP Config>
```

Be aware of the following when you configure a dial circuit that uses a V.25bis interface as the base network:

- The V.25bis interface predefines clocking as external and encoding as NRZ. The modem (DCE) controls the clock speed. You cannot configure clocking, encoding, and other HDLC parameters as part of the dial circuit configuration.

Be aware that you cannot configure HDLC parameters of the dial circuit configuration when you configure PPP or Frame Relay for ISDN. Physical layer parameters are configured on the ISDN interface.

For information on configuring the PPP protocol, refer to Chapter 28, “Configuring Serial Line Interfaces” on page 28-1 or refer to Chapter 33, “Using and Configuring Point-to-Point Protocol Interfaces” on page 33-1.

For information on configuring the Frame Relay protocol, see Chapter 31, “Using and Configuring Frame Relay Interfaces” on page 31-1 or Chapter 32, “Monitoring Frame Relay Interfaces” on page 32-1.

For more information on configuring dial-in and dial-out interfaces, see Chapter 37, “Using and Configuring a Dial-In Access to LANs (DIALs) Server” on page 37-1

To return to the `Circuit Config>` prompt, use the **exit** command.

List

Use the **list** command to display the current dial circuit configuration.

Syntax: list

Example: `list`

```
Base net:       1
Destination name:    remote-site-baltimore
Inbound dst name:    local-1
Outbound calls       allowed
Inbound calls        allowed
Idle timer           = 60 sec
SelfTest Delay Timer = 0 ms
```

Base net:	Name of the serial line interface to which this dial circuit is mapped.
Destination name:	Network address name to be called for outbound circuits, and the default comparison address used by the caller-ID mechanism for inbound calls.
Inbound dst name:	This parameter appears only if the circuit is configured to accept inbound calls that do not match any other addresses.
Inbound dst name:	Alternate comparison address name used by the caller-ID mechanism for inbound calls.
Outbound calls allowed	Displays this parameter when the circuit is configured to initiate outbound calls.
Inbound calls allowed	Displays this parameter when the circuit is configured to accept inbound calls.
Idle timer	Displays the idle timer setting in seconds. The range is 0 to 65535; 0 indicates that this is a dedicated circuit (leased line).
SelfTest Delay Timer	Displays the self-test delay timer setting in milliseconds. The range is 0 to 65535; 0 indicates no delay.

Set

Use the **set** command to map the dial circuit to an interface (eg. ISDN or V.25bis), configure the dial circuit for inbound and/or outbound calls, and set destination addresses, inbound addresses, idle timeout, and self-test delay.

Syntax: set
net...
calls...
destination...
inbound destination...
any_inbound
idle...
lid_out_addr...
priority...
selftest-delay...

net *#*

Sets the base circuit number to *#* of serial line interface to which you want to map this circuit.

Note: The interface must be a V.34 net for dial-out interfaces.

Example:

```
Circuit Config> set net
Base net for this circuit [ ]? 2
```

calls *outbound* or *inbound* or *both*

Restricts this dial circuit to initiating outbound calls only, accepting inbound calls only, or both initiating and accepting calls. The default is both.

Note: If you are using this circuit for WAN-Restoral or another dial-on-demand application, you should set the calls to either **inbound** or **outbound**. This avoids a conflict if both ends of the circuit attempt to establish a call at the same time.

Example: `set calls outbound`

destination *address name*

This parameter is required for the dial circuit to operate. It specifies the network dial address of the remote router to which this dial circuit will connect. The caller-ID protocol uses this parameter as the default comparison address for incoming calls. This parameter must match an address name that you assigned using the `Config>` prompt using either the **add isdn address** command or the **add v25-bis address** command.

Example: `set destination remote-site-baltimore`

inbound destination *address name*

Set this parameter if the dial circuit is set up for both inbound and outbound calls and if this router's local dial address is different from the destination dial address that the remote router dials. For example, the numbers would be different if one of the routers must go through a PBX, international, or inter-LATA exchange. This parameter overrides the default comparison address that the caller-ID protocol uses for incoming calls. This parameter must match an address name that you assigned at the `Config>` prompt using either the **add isdn address** command or the **add v25-bis address** command.

Example: `set inbound remote-site-1`

any_inbound

Specifies that inbound calls that do not match any other dial circuit will be mapped to this circuit and accepted as inbound calls.

Example: `set any_inbound`

idle *# of seconds*

Specifies a timeout period for the circuit. If there is no protocol traffic over the circuit for this specified time period, the dial circuit hangs up. The range is 0 to 65535, and the default is 60 seconds. A setting of zero specifies that there is no timeout period and that this is a dedicated circuit (leased line).

Note: For WAN Restoral operations, you must set the idle timeout to 0.

Example: `set idle 6`

lid_out_addr

The lid_out_addr is the name of a dial circuit between two routers. When more than one circuit is configured between two routers (parallel circuits), then there needs to be a way to unambiguously know which dial circuit connects between them. For this purpose, a lid_out_addr is sent from the router at one end (the caller). At the receiving end the other router configures the same string as the inbound destination name. The lid_out_addr must be an address name that has previously been added using **ADD ISDN-ADDRESS** from the config> prompt.

priority

The priority field allows an outbound dial-on-demand circuit to preempt another when no channels are available. If a call request is made and all the channels are in use, then the priority of the requesting dial-on-demand circuit is checked against all the active dial-on-demand circuits. If there is an outbound dial-on-demand circuit with lower priority, then that circuit is disconnected and a call is made for the higher priority dial-on-demand circuit. Only the priority on the outbound end of a connection is considered. An inbound dial-on-demand call will not be taken down in favor of a higher priority outbound call. An inbound dial-on-demand call cannot cause a lower priority call to be taken down.

selftest-delay *# of milliseconds*

Use this parameter to delay the time between when the call is established and the time when the initial packet is sent. Setting a *selftest-delay* can prevent initial packets from being dropped. The range is 0 to 65535, and the default is 150.

For V.25bis dial circuits, adjust this setting if your modems take extra time to synchronize.

For ISDN dial circuits, you may need to adjust this setting for dial-on-demand links because some ISDN switches start to deliver data before signalling the complete establishment of the circuit at the destination end.

Exit

Use the **exit** command to return to the `Config>` prompt.

Syntax: exit

Example: **`exit`**

Chapter 50. Using and Configuring Quality of Service (QoS)

This chapter describes how to configure the Quality of Service (QoS) feature in the device.

Quality of Service Overview

The QoS feature leverages the benefits of ATM QoS capabilities for LAN Emulation Data Direct VCCs. This support is referred to as “Configurable QoS for LAN Emulation.” The key attributes and the benefits of this feature are as follows:

- An LE Client makes use of configured QoS parameters for its Data Direct VCCs.
- QoS parameters can be configured for:
  - LE Client
  - ATM Interface
- The set of QoS parameters configured are for use with ATM Forum UNI 3.0/3.1 signaling. The parameters include the desired Peak Cell Rate, Sustained Cell Rate, QoS Class and Maximum Burst Size.
- Maximum Reserved Bandwidth per VCC can be configured to protect an LE Client from accepting/establishing VCCs whose traffic parameters it cannot support.
- The QoS Negotiation mechanism enables the participating LE Clients to be aware of each other’s QoS parameters. A data-direct VCC is set up using the negotiated parameters.

Benefits of QoS

- The configured QoS parameters for the LE Client, ATM Interface, or Emulated LAN enable QoS for LANE Data Direct VCCs.
  - An LE Client can be configured with QoS if the QoS required by the client is different from the QoS required by other clients on the ELAN. For example, if an LE Client serves a file server, then the user may want to configure appropriate QoS parameters for all traffic to and from the file server.
  - An ATM Interface can be configured with QoS if a user want all LE Clients on that ATM interface to use the same set of parameters. For example, if an ATM Interface is connected at 25 Mbps, the user can configure appropriate parameters that are different from those at a 155-Mbps interface.

QoS Configuration Parameters

This section describes nine parameters that are used for QoS configuration. The following six parameters can be configured for an LE Client, ATM Interface, and an Emulated LAN:

1. max-reserved-bandwidth
2. traffic-type
3. peak-cell-rate

4. sustained-cell-rate
5. max-burst-size
6. qos-class

The following two parameters can be configured for an Emulated LAN and an LE Client:

1. validate-pcr-of-best-effort-vccs
2. negotiate-qos

The following parameter can be configured only for an LE Client:

accept-qos-parms-from-lecs

The first six parameters control the traffic characteristics of Data Direct VCCs established by the LE Client while the first parameter also applies to the calls received by the LE Client. The following characteristics are associated with all the Data Direct VCCs established by the LE Client:

- Bandwidth is not reserved for best-effort traffic,
- Traffic parameters apply to both forward and backward directions,
- When a reserved bandwidth connection is rejected due to the traffic parameters or QoS Class, the call is retried as a best-effort connection with the configured peak cell rate (cause codes on release or release-complete messages are used to determine why a VCC was released).
- When a best-effort connection is rejected due to the Peak Cell Rate (PCR), the call may be automatically retried with a lower PCR. Retries are performed under the following conditions:
    1. If the rejected PCR is greater than 100 Mbps, the call is retried with a PCR of 100 Mbps.
    2. Otherwise, if the rejected PCR is greater than 25 Mbps, the call is retried with a PCR of 25 Mbps.

Maximum Reserved Bandwidth (max-reserved-bandwidth)

The maximum reserved bandwidth acceptable for a Data Direct VCC. This parameter applies to both Data Direct VCC calls received by the LE Client and Data Direct VCC calls placed by the LE Client. For incoming calls, this parameter defines the maximum acceptable SCR for a Data Direct VCC. If SCR is not specified on the incoming call, then this parameter defines the maximum acceptable PCR for a Data Direct VCC with reserved bandwidth.

Calls received with traffic parameters specifying higher rates will be released. If SCR is specified on the incoming call, the call will not be rejected due to the PCR or Maximum Burst Size. The constraint imposed by this parameter is not applicable to BEST_EFFORT connections. For outgoing calls, this parameter sets an upper bound on the amount of reserved bandwidth that can be requested for a Data Direct VCC. Therefore the traffic-type and sustained-cell-rate parameters are dependent upon this parameter.

Valid Values: Integer in the range 0 to the line speed of ATM device in Kbps
Default Value: 0

Traffic Type (traffic-type)

The desired traffic type for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the type of calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired type of traffic characteristics for Data Direct VCCs. When QoS parameters are negotiated, if either the source or target LEC desires a reserved bandwidth connection and both LECs support reserved bandwidth connections (that is, max-reserved-bandwidth > 0), then an attempt will be made to establish a reserved bandwidth Data Direct VCC between the two LECs. Otherwise, the Data Direct VCC will be a best-effort connection. Dependencies: max-reserved-bandwidth

Valid Values: BEST_EFFORT or RESERVED_BANDWIDTH
Default: BEST EFFORT.

Peak Cell Rate (peak-cell-rate)

The desired peak cell rate for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the PCR traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired PCR traffic parameter for Data Direct VCCs. The minimum of the desired PCRs of the two LECs is used for negotiated best-effort VCCs.

When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired PCR of that LEC is used for the Data Direct VCC subject to the upper bound imposed by the line rate of the local ATM device. If both LECs request a reserved bandwidth connection, then the maximum of the desired PCRs of the LE Clients is used for the Data Direct VCC subject to the upper bound imposed the line rate of the local ATM device.

Valid Values: An integer value in the range 0 to the line speed of ATM device in Kbps
Default Value: Line speed of LEC ATM Device in Kbps.

Sustained Cell Rate (sustained-cell-rate)

The desired sustained cell rate for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the SCR traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired SCR traffic parameter for Data Direct VCCs.

When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired SCR of that LEC is used for the Data Direct VCC (subject to the upper bound imposed by the max-reserved-bandwidth parameter of the other LEC). If both LECs request a reserved bandwidth connection, then the maximum of the desired SCRs of the LE Clients is used for the Data Direct VCC (subject to the upper bound imposed by the max-reserved-bandwidth parameters of both LECs). In any case (negotiation or not), if the SCR that is to be signaled equals the PCR that is to be signaled, then the call is signaled with PCR only.

Dependencies: max-reserved-bandwidth, traffic-type and peak-cell-rate. This parameter is applicable only when traffic-type is RESERVED_BANDWIDTH.

Valid Values: An integer value in the range 0 to the minimum of max-reserved-bandwidth and peak-cell-rate, specified in Kbps

Default Value None

Maximum Burst Size (max-burst-size)

The desired maximum burst size for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the Maximum Burst Size traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired Maximum Burst Size traffic parameter for Data Direct VCCs.

When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired Maximum Burst Size of that LEC is used for the Data Direct VCC. If both LECs request a reserved bandwidth connection, then the maximum of the desired Maximum Burst Sizes of the LE Clients is used for the Data Direct VCC.

In any case (negotiation or not), the Maximum Burst Size is signaled only when SCR is signaled. Although this parameter is expressed in units of cells, it is configured as an integer multiple of the Maximum Data Frame Size (specified in LEC's C3 parameter) with a lower bound of 1.

Dependencies: This parameter is applicable only when traffic-type is RESERVED_BANDWIDTH.

Valid Values: An integer number of frames; must be greater than 0
Default: 1 frame

QoS Class (qos-class)

The desired QoS class for reserved bandwidth calls. If QoS parameters are not negotiated, then this parameter specifies the QoS Class to be used for reserved bandwidth Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the QoS Class that is desired for Data Direct VCCs. Unspecified QoS Class is always used on best-effort calls. Specified QoS Classes define objective values for ATM performance. Specified QoS Classes define objective values for ATM performance parameters such as cell loss ratio and cell transfer delay.

The UNI Specification states that:

Specified QoS Class 1	should yield performance comparable to current digital private line performance.
Specified QoS Class 2	is intended for packetized video and audio in teleconferencing and multimedia applications.
Specified QoS Class 3	is intended for interoperation of connection oriented protocols, such as frame relay.
Specified QoS Class 4	is intended for interoperation of connectionless protocols, such as IP or SMDS.

LECs must be able to accept calls with any of the above QoS Classes. When QoS parameters are negotiated, the configured QoS Classes of the two LECs are compared, and the QoS Class with the more stringent requirements is used.

Valid Values: 0: for Unspecified QoS Class
1: for Specified QoS Class 1
2: for Specified QoS Class 2
3: for Specified QoS Class 3
4: for Specified QoS Class 4
Default Value: 0 (Unspecified QoS Class)

Validate PCR of Best-Effort VCCs (validate-pcr-of-best-effort-vccs)

To validate Peak Cell Rate of Best-Effort VCCs. When FALSE, best-effort VCCs will be accepted without regard to the signaled forward PCR. When TRUE, best-effort VCCs will be rejected if the signaled forward PCR exceeds the line rate of the LE Client ATM device. Calls will not be rejected due to the backward PCR. The signaled backward PCR will be honored if it does not exceed the line rate; otherwise, transmissions to the caller will be at line rate.

Notes:

1. Accepting best-effort VCCs with forward PCRs that exceed the line rate can result in poor performance due to excessive retransmissions; however, rejecting these VCCs can result in interoperability problems.
2. The YES setting is useful when callers will retry with a lower PCR following call rejection due to unavailable cell rate.

Valid Values: YES,NO
Default Value: NO

Negotiate QoS (negotiate-qos)

Enable QoS parameter negotiation for Data Direct VCCs. This parameter should be enabled only when connecting to an IBM MSS LES. When this parameter is YES, the LE Client will include an IBM Traffic Parameter TLV in LE_JOIN_REQUEST and LE_ARP_RESPONSE frames sent to the LES. This TLV will include the values of max-reserved-bandwidth, traffic-type, peak-cell-rate, sustained-cell-rate, max-burst-size and qos-class. An IBM Traffic Parameter TLV may also be included in a LE_ARP_RESPONSE returned to the LE Client by the LES.

If there is no TLV in a LE_ARP_RESPONSE received by the LE Client, then the local configuration parameters must be used to setup the Data Direct VCC. If a TLV is included in a LE_ARP_RESPONSE, the LE Client must compare the contents of the TLV with the corresponding local values to determine the “negotiated” or “best” set of parameters acceptable to both parties before signalling for the Data Direct VCC.

Valid Values: YES,NO
Default Value: NO

Accept QoS Parms from LECS (accept-qos-parms-from-lecs)

This parameter gives the ability to configure an LE Client to accept/reject QoS parameters from a LECS. When this parameter is YES, the LE Client should use the QoS parameters obtained from the LE Clients in the LE_CONFIGURE_RESPONSE frames, that is, the QoS parameters from the LE Clients override the locally configured QoS parameters. If this parameter is NO then

the LE Client will ignore any QoS parameters received in an LE_CONFIGURE_RESPONSE frame from the LE Clients.

Valid Values: YES,NO
Default Value: YES

Accessing the QoS Configuration Prompt

Use the **feature** command from the CONFIG process to access the Quality of Service configuration commands. Enter **feature** followed by the feature number (6) or short name (QOS). For example:

```
Config> feature qos
Quality of Service - Configuration
QoS Config>
```

Once you access the `QoS Config>` prompt, you can configure the Quality of Service (QoS) of an LE Client, or an ATM Interface. To return to the `Config>` prompt at any time, enter the **exit** command at the `QoS Config>` prompt.

Alternatively, you can configure QoS parameters for an LE Client or an ATM Interface by accessing the entities as follows:

- LE Client
  1. At the `Config>` prompt, enter the **network** command and the LE Client interface number.
  2. At the `LE Client configuration>` prompt enter **qos-configuration**.

     Example:

     ```
     config> network 3
     Token Ring Forum Compliant LEC Config> qos-configuration
     LEC QoS Config>
     ```

- ATM Interface
  1. at the `Config>` prompt, enter the **network** command and the ATM interface number to get you to the `ATM Config>` prompt.
  2. Enter the **interface** parameter to get to the `ATM Interface Config>` prompt.
  3. At the `ATM InterfaceConfig>` prompt enter **qos-configuration**.

     Example:

     ```
     config> network 0
     ATM Config> interface
     ATM Interface Config> qos-configuration
     ATM-I/F 0 QoS>
     ```

Quality of Service Commands

This section summarizes and explains the QoS configuration commands. Use the following commands to configure Quality of Service. Enter the commands from the `QoS Config>` prompt.

Table 50-1. Quality of Service (QoS) Configuration Command Summary	
Command	Function
? (Help)	Displays all the QoS configuration commands.
LE-CLIENT	Gets you to the `LE Client QoS configuration >` prompt for the selected LE client.
ATM-INTERFACE	Gets you to the `ATM Interface QoS configuration>` prompt for the selected ATM interface.
Exit	Exits the QoS configuration process.

LE Client QoS Configuration Commands

This section summarizes and explains the commands for configuring QoS for a specific LE Client.

Use the following commands at the `LEC QoS config>` prompt.

Table 50-2. LE Client Quality of Service (QoS) Configuration Command Summary	
Command	Function
? (Help)	Displays all the LE Client QoS configuration commands, or displays the parameters associated with specific commands.
List	Lists the current QoS configuration of the LE Client.
Set	Sets the QoS parameters of the LE Client.
Remove	Removes the QoS configuration of the LE Client.
Exit	Returns to the previous prompt level.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?
Example: **?**

```
list
set
remove
exit
```

List

Use the **list** command to list the QoS configuration of this LE Client. QoS parameters are listed only if at least one has been specifically configured (see Example 1). Otherwise, no parameters are listed (see Example 2).

Syntax: list

Example 1:

```
LEC QoS Config> list

          LE Client QoS Configuration for Data Direct VCCs
          ========================================
     (ATM interface number = 0,  LEC interface number = 3)

  Maximum Reserved Bandwidth for a Data-Direct VCC = 10000 Kbps
  Data-Direct VCC Type ........................... = Best-Effort
  Data-Direct VCC Peak Cell Rate ................. = 155000 Kbps
  Data-Direct VCC Sustained Cell Rate ............ = 155000 Kbps
  Desired QoS Class of Reserved Connections ...... = 0
  Max Burst Size of Reserved Connections    ...... = 0 frames

  Validate Peak Rate of Best-Effort connections .. = No
  Enable QoS Parameter Negotiation ............... = Yes
  Accept QoS Parameters from LECS ................ = Yes

LEC QoS Config>
```

Example 2:

```
LEC QoS Config> list

   QoS has not been configured for this LEC.
   Please use the SET option to configure QoS.

 LEC QoS Config>
```

Set

Use the **set** command to specify LE Client QoS parameters.

Syntax: set all-default-values
max-reserved-bandwidth
traffic-type
peak-cell-rate
sustained-cell-rate
max-burst-size
qos-class
validate-pcr-of-best-effort-vccs
negotiate-qos
accept-qos-parms-from-lecs

all-default-values

Use this option to set the QoS parameters to default values. In the following example the default values are also listed.

Example:

```
LEC QoS Config> set all-default-values
   Failed to locate existing QoS configuration record!
   Using a new set of default values ...
   Initializing all parameters to default values
   LEC QoS Config> list

              LE Client QoS Configuration for Data Direct VCCs
              ========================================
         (ATM interface number = 0,  LEC interface number = 3)

     Maximum Reserved Bandwidth for a Data-Direct VCC = 0 Kbps
     Data-Direct VCC Type ........................... = Best-Effort
     Data-Direct VCC Peak Cell Rate ................. = 155000 Kbps
     Data-Direct VCC Sustained Cell Rate ............ = 155000 Kbps
     Desired QoS Class of Reserved Connections ...... = 0
     Max Burst Size of Reserved Connections    ...... = 0 frames

     Validate Peak Rate of Best-Effort connections .. = No
     Enable QoS Parameter Negotiation ............... = No
     Accept QoS Parameters from LECS ................ = Yes

  LEC QoS Config>
```

max-reserved-bandwidth

Use this option to set the maximum reserved bandwidth allowable per Data Direct VCC. See “Maximum Reserved Bandwidth (max-reserved-bandwidth)” on page 50-2 for a more detailed description of this parameter.

Valid Values: Integer in the range 0 to the line speed of ATM device in Kbps

Default Value: 0

Example:

```
   LEC QoS Config> set max-reserved-bandwidth
   Maximum reserved bandwidth acceptable for a data-direct VCC (in Kbps) [0]? 20000
   LEC QoS Config>
```

traffic-type

Sets the desired traffic for Data Direct VCCs. See “Traffic Type (traffic-type)” on page 50-3 for a more detailed description of this parameter.

Valid Values: BEST_EFFORT or RESERVED_BANDWIDTH

Default: BEST EFFORT.

Example:

```
   LEC QoS Config>
set traffic-type
     Choose from:
        (0): Best-Effort
        (1): Reserved-Bandwidth
   Data Direct VCC Type [0]? 1
   NOTE: Peak Cell Rate has been reset to 1
         Sustained Cell Rate has been reset to 1
         Max Reserved Bandwidth has been reset to 1
         Please configure appropriate values.
   LEC QoS Config>
```

peak-cell-rate

Sets the desired peak cell rate for Data Direct. See “Peak Cell Rate (peak-cell-rate)” on page 50-3 for a more detailed description of this parameter.

Valid Values: An integer value in the range 0 to the line speed of ATM device in Kbps

Default Value: Line speed of LEC ATM Device in Kbps.

Example:

```
LEC QoS Config> set peak-cell-rate
Data-Direct VCC Peak Cell Rate in Kbps [1]? 25000
LEC QoS Config>
```

sustained-cell-rate

Sets the desired sustained cell rate for Data Direct VCCs. See “Sustained Cell Rate (sustained-cell-rate)” on page 50-3 for a more detailed description of this parameter.

Valid Values: An integer value in the range 0 to the minimum of max-reserved-bandwidth and peak-cell-rate, specified in Kbps
Default Value None

Example:

```
LEC QoS Config> se sus
Data-Direct VCC Sustained Cell Rate in Kbps [1]? 10000
LEC QoS Config>
```

max-burst-size

Sets the desired maximum burst size in frames. See “Maximum Burst Size (max-burst-size)” on page 50-4 for a more detailed description of this parameter.

Valid Values: An integer number of frames; must be greater than 0
Default: 1 frame

Example:

```
LEC QoS Config> se ma
Maximum Burst Size in Kbps [1]? 10000
LEC QoS Config>
```

qos-class

Sets the desired QoS Class for Data Direct VCCs. See “QoS Class (qos-class)” on page 50-4 for a more detailed description of this parameter.

Valid Values: 0: for Unspecified QoS Class

1: for Specified QoS Class 1

2: for Specified QoS Class 2

3: for Specified QoS Class 3

4: for Specified QoS Class 4

Default Value: 0 (Unspecified QoS Class)

Example:

```
LEC QoS Config> se qos
Desired QoS Class for Data Direct VCCs [0]? 1
LEC QoS Config>
```

validate-pcr-of-best-effort-vccs

Use this option to enable/disable validation of the Peak Cell Rate traffic parameter of the Data Direct VCC calls received by this LE Client. See “Validate PCR of Best-Effort VCCs (validate-pcr-of-best-effort-vccs)” on page 50-5 for a more detailed description of this parameter.

Valid Values: YES,NO
Default Value: NO

Example:

```
LEC QoS Config> se val y
LEC QoS Config>
```

negotiate-qos

Use this option to enable/disable the LE Client's participation in QoS negotiation. See “Negotiate QoS (negotiate-qos)” on page 50-5 for a more detailed description of this parameter.

Valid Values: YES,NO
Default Value: NO

Example:

```
LEC QoS Config> se neg y
LEC QoS Config>
```

accept-qos-parms-from-lecs

Use this option to enable/disable the LE Client to accept/reject the QoS parameters received from an LECS as TLVs. See “Accept QoS Parms from LECS (accept-qos-parms-from-lecs)” on page 50-5 for a more detailed description of this parameter.

Valid Values: YES,NO
Default Value: YES

Example:

```
elan-x LEC QoS Config> se acc y
elan-x LEC QoS Config>
```

Remove

Use the **remove** command to remove the QoS configuration of this LE Client.

Syntax: **remove**

Example:

```
LEC QoS Config> remove
WARNING: This option deletes the QoS configuration.
         To re-configure use any of the SET options.
Should the LEC QoS configuration be deleted? [No]: yes
Deleted QoS configuration successfully
LEC QoS Config>
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: **exit**

Example 1:

```
LEC QoS Config> exit
QoS Config>
```

Example 2:

```
LEC QoS Config> exit
Token Ring Forum Compliant LEC Config>
```

ATM Interface QoS Configuration Commands

Table 50-3. LE Client Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the ATM Interface QoS configuration commands, or displays the parameters associated with specific commands.
List	Lists the current ATM Interface QoS configuration.
Set	Sets the ATM Interface QoS parameters.
Remove	Removes the QoS configuration of the ATM Interface.
Exit	Returns to the previous prompt level.

List

Use the **list** command to list the QoS configuration of this ATM Interface. QoS parameters are listed only if at least one parameter has been configured (see following example). Otherwise, no parameters are listed.

Syntax: list

Example:

```
ATM-I/F 0 QoS> list

          ATM Interface  Quality of Service' Configuration
          ================================================
                    (ATM interface number = 0  )

     Maximum Reserved Bandwidth for a VCC = 15000 Kbps
     VCC Type ........................... = RESERVED-BANDWIDTH
     Peak Cell Rate ..................... = 20000 Kbps
     Sustained Cell Rate ................ = 5000 Kbps
     QoS Class .......................... = 4
     Maximum Burst Size ................. = 5 frames
ATM-I/F 0 QoS>
```

Set

Use the **set** command to specify ATM Interface QoS parameters.

Syntax: set max-reserved-bandwidth

traffic-type

peak-cell-rate

sustained-cell-rate

max-burst-size

qos-class

max-reserved-bandwidth

Use this option to set the maximum reserved bandwidth allowable for each Data Direct VCC. See “Maximum Reserved Bandwidth (max-reserved-bandwidth)” on page 50-2 for a more detailed description of this parameter.

Valid Values: Integer in the range 0 to the line speed of ATM device in Kbps

Default Value: 0

Example:

```
   ATM-I/F 0 QoS> se max-reserved-bandwidth
   Maximum reserved bandwidth acceptable for a data-direct VCC (in Kbps) [0]?
15000
   ATM-I/F 0 QoS>
```

traffic-type

Sets the desired traffic for Data Direct VCCs. See “Traffic Type (traffic-type)” on page 50-3 for a more detailed description of this parameter.

Valid Values: BEST_EFFORT or RESERVED_BANDWIDTH
Default: BEST EFFORT.

Example:

```
ATM-I/F 0 QoS> set traffic-type
  Choose from:
     (0): Best-Effort
     (1): Reserved Bandwidth
Traffic Type of VCCs [1]? 0
ATM-I/F 0 QoS>
```

peak-cell-rate

Sets the desired peak cell rate for Data Direct VCCs. See “Peak Cell Rate (peak-cell-rate)” on page 50-3 for a more detailed description of this parameter.

Valid Values: An integer value in the range 0 to the line speed of ATM device in Kbps
Default Value: Line speed of LEC ATM Device in Kbps.

Example:

```
ATM-I/F 0 QoS Config> set peak-cell-rate
Data-Direct VCC Peak Cell Rate in Kbps [1]? 25000
ATM-I/F 0 QoS Config>
```

sustained-cell-rate

Sets the desired sustained cell rate for Data Direct VCCs. See “Sustained Cell Rate (sustained-cell-rate)” on page 50-3 for a more detailed description of this parameter.

Valid Values: An integer value in the range 0 to the minimum of max-reserved-bandwidth and peak-cell-rate; specified in Kbps
Default Value None

Example:

```
ATM-I/F 0 QoS Config> se sus
Data-Direct VCC Sustained Cell Rate in Kbps [1]? 10000
ATM-I/F 0 QoS Config>
```

max-burst-size

Sets the desired maximum burst size in frames. See “Maximum Burst Size (max-burst-size)” on page 50-4 for a more detailed description of this parameter.

Valid Values: An integer number of frames; must be greater than 0
Default: 1 frame

Example:

```
ATM-I/F 0 QoS Config> se ma
Maximum Burst Size in Kbps [1]? 10000
ATM-I/F 0 QoS Config>
```

qos-class

Sets the desired QoS Class for Data Direct VCCs. See “QoS Class (qos-class)” on page 50-4 for a more detailed description of this parameter.

Valid Values: 0: for Unspecified QoS Class

1: for Specified QoS Class 1

2: for Specified QoS Class 2

3: for Specified QoS Class 3

4: for Specified QoS Class 4

Default Value: 0 (Unspecified QoS Class)

Example:

```
ATM-I/F 0 QoS Config> se qos
Desired QoS Class for Data Direct VCCs [0]? 1
ATM-I/F 0 QoS Config>
```

Remove

Use the **remove** command to remove the QoS configuration of this ATM Interface.

Syntax: remove Example:

```
ATM-I/F 0 QoS> remove
WARNING: This option deletes the QoS configuration.
         To re-configure use any of the SET options.
Should the ATM Interface QoS configuration be deleted? [No]: yes
Deleted QoS SRAM record successfully
ATM-I/F 0 QoS>
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: **exit**

Example 1:

```
ATM-I/F 0 QoS> exit
QoS Config>
```

Example 2:

```
ATM-I/F 0 QoS> exit
ATM Interface Config>
```

Chapter 51. Monitoring Quality of Service (QoS)

This chapter describes how to monitor Quality of Service (QoS) for LAN and ELAN interfaces in the router. It contains the following sections:

Accessing the QoS Console Commands

Use the **feature** command from the GWCON process to access the Quality of Service console commands. Enter the **feature** followed by the feature number (6) or short name (QOS). For example:

```
+feature qos
Quality of Service (QoS) - User Console
QoS+
```

Once you access the QoS console prompt, you can select the console of a particular LE Client. To return to the GWCON prompt at any time, enter the exit command at the QoS console prompt.

Alternatively, you can access the QoS Console of an LE Client as follows:

1. At the GWCON prompt (+), enter the network command and the LE Client interface number.
2. At the LE Client console prompt enter **qos-information**.

   Example:

```
+network 3
ATM Emulated LAN Console
LEC+qos information
LE Client QoS Console
LEC 3 QoS+
```

Quality of Service Console Commands

This section summarizes and explains the QoS console commands. Enter these commands at the QoS+ prompt.

Table 51-1. Quality of Service (QoS) Console Command Summary

Command	Function
? (Help)	Displays all the QoS console commands.
LE-CLIENT	Gets you to the `LE Client QoS console +` prompt for the selected LE client.
Exit	Exits the QoS console process.

LE Client QoS Console Commands

This section summarizes and explains the LE Client QoS console commands. Enter the commands from the `LEC num QoS+` prompt.

Table 51-2. LE Client QoS Console Command Summary

Command	Function
? (Help)	Displays all the LE Client QoS console commands or displays the options associated with specific commands.
List	Lists the current LE Client QoS information. Options include: configuration parameters, TLVs, VCCs, and statistics.
Exit	Returns to the previous prompt level.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?
Example: **?**

```
list
exit
```

List

Use the list command to list the QoS related information of this LE Client.

Syntax: list configuration-parameters

tlv-information

vcc-information

data-direct-VCCs (Detailed Information)

statistics

configuration-parameters

Lists the QoS configuration parameters. Because parameters can be configured for an LE Client, ATM Interface or the ELAN, these parameters are displayed along with a resolved set of parameters that are used by the LE Client.

le-client The parameters configured for this LE Client which are obtained from the SRAM records. If the SRAM records contain an invalid set of parameters then this column will not display any parameters values.

ATM Interface The parameters configured for the ATM Interface used by this LE Client. These parameters are obtained from the local SRAM records. If the SRAM records contain an invalid set of parameters then this column will not display any parameter values.

From LECS The parameters received by this LE Client from the LE Configuration Server. The parameters are received as individual TLVs in the LE_CONFIGURE_RESPONSE control message.

used The resolved set of traffic parameters which are used by for its Data Direct VCCs. If none of the entities is configured with QoS parameters, then the USED parameters represent the default parameters. If parameters are configured for at least one entity, then they are resolved as follows:

- If only the LE Client or the ATM Interface is configured with parameters and either the accept-parms-from-lecs is FALSE or no parameters were received from the LECS, then the configured LE Client or the ATM Interface parameters are used.
- If both the LE Client and the ATM Interface have configured parameters, then the LE Client parameters are used.
- If the accept-parms-from-lecs is TRUE and parameters were received from the LECS, then the LE Client parameters (or the default if the LE Client is not configured) are combined with those received from the LECS to form a complete set of the first six QoS parameters described in “QoS Configuration Parameters” on page 50-1.
- If the set of the first six QoS parameters described in “QoS Configuration Parameters” on page 50-1 contains an invalid combination then the parameters from the LECS are rejected. Note that the two flags negotiate-qos and validate-pcr-of-best-effort-vccs are validated independently.

Example:

```
LEC 1 QoS+ list configuration parameters


                ATM LEC Configured QoS  Parameters
                ==================================

     QoS                                        |    LEC      ATM-IF    FROM

   PARAMETER                             USED    |    SRAM     SRAM      LECS

 ------------------------------------------------------------------------------
 Max Reserved Bandwidth (cells/sec) :    23584   |    23584        0     none
                       (Kbits/sec) :    10000   |    10000        0     none
 VCC Type ........................ :   ResvBW   |   ResvBW   BstEft        0
 Peak Cell Rate ........(cells/sec) :    18867   |    18867   365566   365566
                       (Kbits/sec) :     8000   |     8000   155000   155000
 Sustained Cell Rate ...(cells/sec) :    18867   |    18867   365566     none
                       (Kbits/sec) :     8000   |     8000   155000     none
 QoS Class ....................... :        4   |        4        0     none
 Max Burst Size .............(cells) :       95   |       95        0     none
                          (frames) :        1   |        1        0     none
 Validate PCR of Best-Effort VCCs . :       NO   |       NO      n/a     none
 Enable QoS Negotiation ........... :      YES   |      YES      n/a     none
 Accept QoS Parameters from LECS .. :      YES   |      YES      n/a      n/a
 ------------------------------------------------------------------------------
(BstEft = Best Effort, ResvBW = Reserved Bandwidth)
(n/a = not applicable, none = no value is specified)

LEC 1 QoS+
```

tlv-information

Lists the IBM Traffic Information TLV that this LE Client registered with the LE Server. The TLV is registered only if the LE Client is participating in QoS Negotiation.

Example:

```
LEC 1 QoS+ list tlv

  Traffic Info TLV of the LEC (registered with the LES)
  =====================================================
    TLV Type .........................= 268458498
    TLV Length .......................= 24
    TLV Value:
           Maximum Reserved Bandwidth = 23584 cells/sec (10 Mbps)
           Data Direct VCC Type...... = RESERVED BANDWIDTH VCC
           Data Direct VCC PCR....... = 18867 cells/sec (8 Mbps)
           Data Direct VCC SCR....... = 18867 cells/sec (8 Mbps)
           Data Direct VCC QoS Class  = 4
           Maximum Burst Size         = 95 cells (1 frames)

LEC 1 QoS+
```

vcc-information

Lists all active VCCs of the LE Client. The information includes the traffic parameters of the connections. For BEST-EFFORT connections, the Sustained Cell Rate is displayed to be the same as the Peak Cell Rate, QoS Class and the Maximum Burst Size are displayed as 0.

The Parameter Descriptor entries are:

SrcParms Parameters of a connection established by this LE Client.
DestParms Parameters of a connection received by this LE Client.
NegoParms Parameters of a connection established by the LE Client for which the QoS Negotiation was used.
RetryParms Parameters of a connection established by this LE Client after failing at least once.

Example:

```
LEC 1 QoS+ li vcc

                          LEC VCC Table
                          =============

                                                                        Burst
Conn   Conn              Conn          VCC    PCR    SCR   QoS   Size    Parameters
Index  Handle VPI   VCI   Type  Status  Type  (kbps) (kbps) Class (cells) Descriptor
-------------------------------------------------------------------------------------
   2)     69    0   535  Cntrl  Ready  BstEft 155000 155000    0      0   SrcParms
   3)     71    0   537  Cntrl  Ready  BstEft      0      0    0      0   DestParms
   4)     72    0   538  Mcast  Ready  BstEft 155000 155000    0      0   SrcParms
   5)     74    0   540  Mcast  Ready  BstEft      0      0    0      0   DestParms
   6)     78    0   544  Data   Ready  ResvBW  25000   7000    1     95   DestParms

LEC 1 QoS+
```

data-direct-vccs (Detailed Information)

This option lists the Data Direct VCC information of this LE Client. Similar information is also listed using **list vcc-information**.

Example:

```
LEC 1 QoS+ list data direct vccs

          LEC Data Direct VCCs - QoS Information
          ======================================

Conn Handle = 80, VPI = 0, VCI = 546
        Connection Type = RETRIED CONNECTION PARAMETERS
          TrafficType    = BEST EFFORT VCC
          PCR            = 58962 (25 Mbps)
          SCR            = 58962 (25 Mbps)
          QoS Class      = 0
          Max Burst Size = 0

Conn Handle = 78, VPI = 0, VCI = 544
        Connection Type = PARAMETERS SET BY DESTINATION
          TrafficType    = RESERVED BANDWIDTH VCC
          PCR            = 58962 (25 Mbps)
          SCR            = 16509 (7 Mbps)
          QoS Class      = 1
          Max Burst Size = 95

LEC 1 QoS+
```

statistics

Counters are maintained for the following statistics:

Successful QoS Connections	Number of RESERVED-BANDWIDTH connections established by the LE Client.
Successful Best-Effort Connections	Number of BEST-EFORT connections established by the LE Client.
Failed QoS Connections	Number of RESERVED-BANDWIDTH connection requests made by the LE Client that failed.
Failed Best-Effort Connections	Number of BEST-EFFORT connection requests made by the LE Client that failed.
Qos Negotiation Applied	Number of times the QoS negotiation extension was applied. Parameters are negotiated if the LE Client receives the destination LE Client's parameters in an LE_ARP_RESPONSE control message.
PCR Proposal (IBM) Applied	Number of times the IBM Peak Cell Rate Proposal was applied. This proposal recommends using specific rate parameters if signaling at 100 Mbps or 155 Mbps for BEST-EFFORT connections. This allows other participating IBM products (for example, 25-Mbps ATM adapters) to reject a connection based on the signaled peak cell rates.
QoS Connections Accepted	Number of RESERVED-BANDWIDTH connections accepted by this LE Client.
Best-Effort Connections Accepted	Number of BEST-EFFORT connections accepted by this LE Client.
QoS Connections Rejected	Number of RESERVED-BANDWIDTH connection requests received by this LE Client that were rejected.
Best-Effort Connections Rejected	Number of BEST-EFFORT connection requests received by this LE Client that were rejected.

Rejected due to PCR Validation	Number of BEST-EFFORT connections rejected by the LE Client due to validation of Peak Cell Rate when the validate-pcr-of-best-effort-vccs parameter is TRUE.

Example:

```
LEC 1 QoS+ li stat

QoS Statistics: of Data Direct Calls Placed by the LEC
------------------------------------------------------
    Successful QoS Connections          = 0
    Successful Best-Effort Connections  = 1
    Failed QoS Connections              = 1
    Failed Best-Effort Connections      = 1
    Qos Negotiation Applied             = 0
    PCR Proposal (IBM) Applied          = 0

QoS Statistics: of Data Direct Calls Received by the LEC
--------------------------------------------------------
    QoS Connections Accepted            = 1
    Best-Effort Connections Accepted    = 0
    QoS Connections Rejected            = 0
    Best-Effort Connections Rejected    = 0
    Rejected due to PCR Validation      = 0

LEC 1 QoS+
```

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: **exit**

Example:

```
LEC 1 QoS+ exit
QoS+
```

Chapter 52. Using and Configuring ATM

This chapter describes how to configure the ATM interface. It includes the following sections:

ATM and LAN Emulation

LAN emulation provides support for virtual Token-Ring and Ethernet LANs over an ATM network. Refer to "How to Enter Addresses" for a discussion of ATM addressing.

How to Enter Addresses

Enter addresses in two ways, depending upon whether the address represents (1) an IP address, or (2) an ATM address, MAC address, or route descriptor, as follows:

1. IP address

   Enter IP addresses in dotted decimal format, a 4-byte field represented by four decimal numbers (0 to 255) separated by periods (.).

   Example of IP Address:

   ```
   01.255.01.00
   ```

2. ATM or MAC address or route descriptor

   Enter ATM addresses, MAC addresses, and route descriptors as strings of hexadecimal characters with or without optional separator characters between bytes. Valid separator characters are dashes (-), periods (.), or colons (:).

   Examples of ATM address, MAC address or route descriptor

   ```
   A1FF010203
       or
   A1-FF-01-02-03
       or
   A1.FF.01.02.03
       or
   39.84.0F.00.00.00.00.00.00.00.00.00.03.10.00.5A.00.DE.AD.08
       or
   A1:FF:01:02:03
       or even
   A1-FF.01:0203
   ```

   Each type of address requires a different number of hexadecimal characters:

   ATM 40
   MAC 12
   ESI 12
   Route descriptor 4

This information applies to addresses entered for ATM; LAN emulation; Classical IP and ARP over ATM; and IPX and ARP over ATM.

ATM-LLC Multiplexing

Protocols that run natively over an ATM interface can use ATM-LLC multiplexing to share ATM addresses and both SVC and PVC channels between users. ATM-LLC is implicitly configured when the protocols are configured and can be monitored using the `ATM Config+` command prompt from **t 5**. There are no explicit configuration options for the ATM-LLC multiplexing function. For example, if two protocols which use ATM-LLC multiplexing are configured to use the same local ATM address (local endpoint), this implicitly configures ATM-LLC to use the same shared ATM address for both protocols.

See “ATM-LLC Monitoring” on page 53-5 for additional information.

Sharing of ATM addresses or SVC/PVC channels is not possible between protocols that use the ATM-LLC multiplexing function and those that do not use the ATM-LLC multiplexing function (such as Classical IP). Currently, Server Cache Synchronization Protocol (SCSP) and APPN are the only two protocols that use the ATM-LLC multiplexing function.

Accessing the ATM Interface Configuration Process

The ATM carrier card and the 25 Mbps Charm Adapter must be in the feature slot before ATM can be configured. You must reload the router after the feature slot has the ATM carrier card/25 Mbps Charm Adapter combination in place.

Use the following procedure to access the configuration process.

1. At the OPCON prompt, enter **talk 6**. (For more detail on this command, refer to Chapter 2, “The OPCON Process and Commands” on page 2-1.) For example:

   ```
   * talk 6
     Config>
   ```

   The CONFIG prompt (`Config>`) displays on the console. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the CONFIG prompt, enter the **list devices** command to display the network interface numbers for which the router is currently configured.

3. Record the interface numbers.

   If ATM is not specified as an interface, then execute the quick configuration process, *qconfig* to dynamically add the ATM interface.

4. Enter the **network** command and the number of the ATM interface you want to configure. For example:

   ```
   Config> network 0
   ATM Config>
   ```

   The ATM configuration prompt (`ATM Config>`), is displayed.

ATM Configuration Commands

This section summarizes the ATM configuration commands. Enter the commands at the `ATM config>` prompt.

Table 52-1. ATM Configuration Command Summary

Command	Function
? (Help)	Lists all of the ATM configuration commands, or lists the options associated with specific commands.
INTERFACE	Displays the `ATM Interface Config>` prompt from which you can list, change, or configure the ATM Interface. • Add an ESI. • List the current configuration or list ESIs. • Remove an ESI. • Set parameters of the ATM network. • Enable or disable an ESI. • Exit
LE-CLIENT	Displays the `LE Client Config>` prompt from which you can list, change, or configure the LAN Emulation Client Interface as described in Chapter 54, "Using and Configuring LAN Emulation Clients" on page 54-1. • Add a LAN Emulation Client (LEC) for a token-ring or Ethernet emulated LAN. • Configure a LEC by network #. This command displays the `LE Config>` prompt, from which you can configure a specific LAN Emulation Client (LEC). • List LAN Emulation Clients (LECs). • Remove a LAN Emulation Client (LEC).
VIRTUAL ATM	Displays the `ATM Virtual Interface Config>` prompt from which you can list, add, or remove the ATM Virtual Interface as described in "ATM Virtual Interface Configuration Commands" on page 52-11
Exit	Exit the ATM Configuration process and returns to the `Config>` prompt.

ATM Interface Commands

This section summarizes and then explains the commands for configuring a specific ATM interface.

Enter the commands at the `ATM INTERFACE>` prompt.

Table 52-2. ATM INTERFACE Configuration Command Summary

Command	Function
? (Help)	Lists all the ATM Interface configuration commands, or lists the options associated with specific commands.
Add	Adds an ESI.
List	Lists the current configuration or list ESIs.
Qos	Displays the `ATM I/F 0 QOS Config>` prompt from which you can configure Quality of Service as described in "QoS Configuration" on page 52-5.
Remove	Removes an ESI.
Set	Sets parameters of the ATM network.
Disable	Disables an ESI.
Enable	Enables an ESI.
Exit	Returns to the `ATM Config>` prompt.

Add

Use the **add** command to add an ESI to your ATM configuration.

Octets 14–19 of an ATM address are the End System Identifier (ESI). Each end system attached to the same switch must use a disjoint set of ESIs. When a end system activates, it attempts to register its ESIs with its ATM switch using ILMI. The switch forces all of its registered ESIs to be unique.

Syntax: add esi *esi-address*

esi *esi-address*
Address of End System Identifier.

Valid Values: Any 12 hexadecimal digits
Default Value: none

```
ATM INTERFACE> add esi 014617183763
```

List

Use the **list** command to list the configuration of this ATM device or to list the set of configured ESIs.

Syntax: list configuration
esi

configuration
Lists the ATM device configuration. For an explanation of the listed fields, see "Set" on page 52-5.

Example: `list`

```
                        ATM Configuration

  Interface (net) number =     0
  Maximum VCC data rate Mbps   =      155
  Maximum frame size    = 9234
  Maximum number of callers =  209
  Maximum number of calls =  1024
  Maximum number of parties to a multipoint call =  512
  Maximum number of Selectors that can be configured  =  200
  UNI Version = UNI 3.0
  Packet trace = OFF
```

ESIs

Lists the ESIs in the ATM configuration.

Example: `list esi`

```
ATM INTERFACE> list esi

        ESI                Enabled
-----------------------   -------
000000000009                 YES
000000000100                 YES
```

QoS Configuration

Use the **qos-configuration** command to display the `ATM I/F 0 QOS Config>` prompt from which you can configure Quality of Service as described in "QoS Configuration."

Remove

Use the **remove** command to remove an ESI from your ATM configuration. All ATM components using this ESI should be reconfigured to use a different ESI. An ATM component that attempts to use a removed ESI may not activate on the next router restart.

Syntax: remove esi *esi-address*

esi *esi-address*

Address of End System Identifier.

Valid Values: Any 12 hexadecimal digits
Default Value: none

Set

Use the **set** command to specify ATM network parameters.

Syntax: set max-data-rate
max-frame
max-config-selectors
max-calls
max-callers
max-mp-parties
trace
uni-version
network-id

max-data-rate

Sets the default and upper bound for VCC traffic parameters of most LANE and CIP connections. For example, this is the default PCR for best-effort VCCs initiated by LE Clients. Signaled SCRs and PCRs cannot exceed this limit. The default value should be satisfactory in most situations. An example of a situation where it is beneficial to change this value would be if the majority of the stations use 25–Mbps adapters. In this case, it may be desirable to limit the data rate on VCCs to 25 Mbps so that the lower speed stations are not overwhelmed with frames from the router. The units for this parameter are Mbps.

Valid Values: 25
100
155
Default Value: 25

Example:

```
ATM INTERFACE> set speed 25
```

max-calls

Sets the maximum number of switched virtual circuits (SVCs) that can exist on this ATM device. Every point-to-point and point-to-multipoint SVC uses system resources. This parameter helps limit the system resources reserved for signaling and switched connections. Increasing this parameter will allow more simultaneous SVCs. However, more system memory will be required to manage these connections.

Valid Values: An integer in the range 64 – 10500
Default Value: 1024

Example:

```
ATM INTERFACE> set max-calls 500
```

max-callers

Sets the maximum number of entities on the router that use the ATM interface. Each LEC, Classical IP Client, and 1483 bridge interface qualifies as a user of the ATM interface. Increasing this parameter allows more users of the interface and uses more system memory.

Valid Values: An integer in the range 64 – 1024
Default Value: 209

Example:

```
ATM INTERFACE> set max-callers 25
```

max-config-selectors

Sets the maximum number of selectors under your specific control.

The selector is used to distinguish different users on the same end system. VCC setup requests are routed in the following hierarchical fashion: ATM switches route to the destination ATM switch using the Network Prefix, the destination ATM switch routes to the destination end system using the ESI, and the end system notifies the destination user based on the selector.

Each ESI can have up to 255 associated selectors (0x00 through 0xff). The range of selectors is partitioned into two subranges, a configured selector range and an automatically assigned selector range. The ATM interface parameter max-configured-selector gives the upper bound on the configured selector range.

The ATM components on the router have various ways of choosing a selector. Some components require you to explicitly configure a selector from the configured selector range. LES/BUSs are an example of such a component. Other components, such as Classical IP clients, allow the selector to be automatically assigned at run-time. You do not have to choose the selector because the router does this when it activates. This selector is not guaranteed to be consistent across router restarts. Automatic selector assignment is useful only for those ATM components whose ATM address does not have to be already known by other network devices.

The relative sizes of the selector range can be modified to conform to the types and numbers of ATM users on the router.

Valid Values: 0 – 255 (0x00 – 0xFF)
Default Value: 200

Note: The selector is byte 20 of a 20-byte ATM address.

Example:

```
ATM INTERFACE> set max-config-selectors 225
```

max-frame

Sets the maximum number of octets permitted in any data frame sent or received on the ATM interface. System memory is allocated based upon this parameter. Increasing the max-frame requires more system memory, but allows processing of larger frames.

All router entities using the ATM interface must use a maximum frame size less than or equal to the max-frame-size of the ATM interface. This includes all LECsand 1483 bridge interfaces.

Valid Values: An integer in the range 16 – 32000
Default Value: 9234

Example:

```
ATM INTERFACE> set max-frame 1000
```

max-mp-parties

Sets the maximum number of leaves on a point-to-multipoint connection initiated by the router. This parameter affects system memory allocation. Increasing this value is necessary if the router must set up point-to-multipoint connection(s) to a large number of destinations.

Valid Values: An integer in the range 1 – 5000
Default Value: 512

Example:

```
ATM INTERFACE> set max-mp-parties 300
```

Valid Values: 25 (Mbps)
Default Value: 25 (Mbps)

Example:

```
ATM INTERFACE> set speed 25
```

trace

Sets the packet tracing parameters on the interface. Packet tracing can be enabled or disabled on a range of VPI/VCI values. Common VPI/VCI values to trace are:

- 0/5 for signalling packets
- 0/16 for ILMI packets.

Valid Values: ON or OFF
Default Value: ON

You are prompted for the VPI/VCI range you want to trace.

Beginning VPI Valid Values: 0 – 255
Default Value: 0

Ending VPI Valid Values: 0 - 255
Default Value: 255

Beginning VCI Valid Values: 0 - 65535
Default Value: 0

Ending VCI Valid Values: 0 - 65535
Default Value: 65535

Example:

```
ATM INTERFACE> set trace on
beginning of VPI range [0]? 0
end of VPI range [255]? 0
beginning of VCI range [0]? 5
end of VCI range [65535]? 5
```

uni-version

Sets the User Network Interface (UNI) version used by the ATM interface with communicating with the attached ATM switch. The UNI versions used by the ATM interface and by the ATM switch must match. If the UNI version is configured as AUTO, the ATM device attempts to learn the UNI version to use from the switch.

Valid Values: [UNI 3.0|UNI 3.1|AUTO-DETECT|None]
Default Value: UNI 3.0

Note: Must be compatible with the ATM switch.

Example:

```
ATM INTERFACE> set uni-version 3.0
```

network-id

Sets the network id of the ATM interface. Multiple ATM interfaces should have the same network id if there is ATM connectivity between the interfaces.

Valid Values: 0 - 255
Default Value: 0

Enable

Use the **enable** command to enable an ESI in the configuration of your ATM device. The ATM interface attempts to register only enabled ESIs when it activates.

Syntax: enable esi *esi-address*

esi *esi-address*

Address of End System Identifiers.

Valid Values: Any 12 hexadecimal digits
Default Value: none

Example: enable esi

```
ATM INTERFACE> enable esi 00:00:00:00:00:09
```

Disable

Use the **disable** command to disable an ESI in the configuration. ATM components using disabled ESIs will not become active on the next router restart.

Syntax: disable esi *esi-address*

esi *esi-address*

Address of End Sytem Identifiers.

Valid Values: Any 12 hexadecimal digits
Default Value: none

Example: `disable esi`

```
ATM INTERFACE> disable esi 00:00:00:00:00:09
```

Exit

Use the **exit** command to return to the `ATM Config>` prompt.

Syntax: exit

Example: `exit`

```
ATM INTERFACE> exit
ATM Config>
```

ATM Virtual Interface Concepts

An ATM Virtual Interface (AVI) creates the appearance of multiple ATM interfaces when, in fact, there is only one physical ATM interface. One or more AVIs can be configured for each physical ATM interface on the router. AVIs have the following characteristics:

- Each AVI must be defined on one (and only one) physical ATM interface. ATM real interface (ARI) will be used to mean a physical ATM interface.
- One or more AVIs can be configured on each ARI on a router.
- Higher layer protocols treat ARIs and AVIs equally. The protocols see the total number of ATM interfaces as the sum of the number of ARIs and AVIs configured on the router.
- Protocols can be configured on each ATM interface (real or virtual) independently of other interfaces.

  For example, one can configure IP on interface 0 (which is a real ATM interface) with IP address 9.1.1.1 and another instance of IP with address 9.2.1.1 on interface 1 (which is an AVI). Whether an interface is a real ATM interface or a virtual interface configured on a real interface makes no difference to the protocol (IP in the example). In addition, whether virtual interface 1 is configured on top of real ATM interface 0 or some other physical ATM interface is also transparent to the protocols.

Advantages of Using ATM Virtual Interfaces

Major advantages of using the ATM Virtual Interfaces are:

- Using the ATM Virtual Interface feature increases the number of protocol instances that can be supported on a physical ATM interface.

  The actual number of AVIs that can be configured on an ARI is limited by physical resources, such as memory, available on the router. The total number of interfaces that can be created depends on the data packet size for the interfaces and is limited to a maximum number of 253 per router.

  The use of AVIs significantly improves the configuration options for protocols such as IPX that are limited to one instance or address per ATM interface. By configuring an appropriate number of AVIs, several IPX addresses can be supported on each physical ATM interface.
- The ATM Virtual Interface feature is crucial for supporting multicast routing protocols (such as OSPF) over ATM networks.

In order for multicast to operate correctly, each logical subnet ***must*** be configured on a different interface because multicast routing protocols typically function in such a way that a packet coming in from a router interface will never be sent out over the same interface. Thus, if more than one subnet is configured on an interface and a source in one subnet sends a multicast packet to a member in another subnet defined on the same interface, this member will never receive the packet.

By creating an individual virtual interface for each subnet, packet multicasting can be performed successfully. Typically, the number of ATM interfaces on a router will be limited, in turn limiting the number of subnets that can be correctly configured for multicast operation. However, by creating as many AVIs as needed (according to the number of subnets that are required to be configured on the router), the number of physical ATM interfaces will no longer limit the number of subnets that can be configured on a router for correct multicast operation.

For example, the “one-armed” router cannot support multicast traffic over interfaces other than ELANs without the AVI feature, because incoming packets will never be sent out the same interface and will be discarded instead.

- Creating multiple AVIs on an ARI and configuring each different protocol instance (for example, each IP subnet) on a different AVI on the same ARI, can improve performance.

  For example, when multiple subnets are configured on a single physical ATM interface, the interface will have to reduce the maximum transmission unit or MTU (the maximum packet size that can be sent or received over that interface) to the smallest MTU of all subnets sharing the same interface. However, if multiple AVIs are created on that ARI and each IP subnet is configured on a different AVI, every subnet can continue to use its existing MTU size without consideration of other subnets configured on the same physical ATM interface. This avoids possible reduction in throughput and delays due to packet fragmentation and re-assembly caused by MTU size reduction.

  Another performance improvement can be achieved by distributing the number of protocol addresses configured on a physical interface over different virtual interfaces configured on the same physical interface. The per-interface protocol lists get shortened, resulting in faster searches and reduced processing time.

Disadvantages of using ATM Virtual Interfaces

The disadvantages of using ATM Virtual Interfaces are:

- Because AVIs do not have any physical resources of their own, each virtual interface may have fewer Virtual Connections (VCs) than a single physical interface. The available resources (in this example VCs) are partitioned among the different virtual interfaces configured on a single ARI and the ARI itself.

  In the current implementation, resource allocation is *on demand*. Each physical ATM interface has a pool of resources that are available for use by all AVIs and the single ARI itself.

Note: Because all resources are shared among the ARI and all its AVIs, an ESI added on an ARI is automatically available to all AVIs configured on the ARI. You should not assign the same ESI and selector combination to two different protocol clients using the same ARI even though they are configured on different AVIs. Likewise, do not configure different PVCs on

the same ARI with the same VPI/VCI combination even though the PVCs are configured on different AVIs.

Access to the Virtual ATM Interface Configuration

From the `ATM Config>` prompt of a selected real ATM interface, use the **Virtual ATM** command to enter the Virtual ATM configuration command mode.

ATM Virtual Interface Configuration Commands

This section summarizes the ATM virtual interface configuration commands. Enter the commands at the `ATM virtual interface config>` prompt.

Table 52-3. ATM Virtual Interface Configuration Command Summary

Command	Function
? (Help)	Lists all of the ATM virtual interface configuration commands, or lists the options associated with specific commands.
ADD	Adds a virtual ATM interface.
LIST	Lists the current configured virtual ATM interfaces.
REMOVE	Removes the virtual ATM interface from the current configuration.
EXIT	Exit the ATM virtual interface Configuration process and return to the `ATM Config>` prompt.

Help (?)

Use the **help** command to list the configuration options for ATM virtual interfaces.

Syntax: help (or?)

Example: `help`

```
ATM Virtual Interface config> help
ADD
LIST
REMOVE
EXIT
ATM Virtual Interface config>
```

Add

Use the **add** command to add an ATM virtual interface. A new ATM virtual interface is added to the corresponding ATM real interface (the configuration menu from which this ATM virtual interface configuration menu is accessed). The net/interface number assigned to the newly created ATM virtual interface is displayed and it is one number greater than the current largest interface number.

Syntax: add

Example: `add`

```
ATM Virtual Interface config> add
Added ATM Virtual Interface Net as interface 5 on physical ATM interface 0
ATM Virtual Interface config>
```

List

Use the **list** command to list configured ATM virtual interfaces defined on the current real ATM interface.

Syntax: list

Example: `list`

```
ATM Virtual Interface config> list

                  ATM Virtual Interface Nets
------------------------------------------------------------------------
  ATM interface number = 0
  ATM Virtual Interface Net interface number = 5
ATM Virtual Interface config>
```

Remove

Use the **remove** command to delete an ATM virtual interface. The virtual ATM interface on the real ATM interface with the specified interface number will be removed from the SRAM configuration records. If you do not specify an interface number, the last ATM virtual interface on this real ATM interface will be deleted. If you enter a question mark (?), all ATM virtual interfaces on the current real ATM interface will be listed and you can select from that list the interface you want to remove.

Syntax: remove *n*

Example: `remove` 5

```
Virtual ATM 5 deleted successfully.
ATM Virtual Interface config>
```

Exit

Use the **exit** command exit configuration menu for ATM virtual interfaces defined on the current real ATM interface and return to the real ATM interface configuration prompt.

Syntax: exit

Example: `exit`

```
ATM Virtual Interface config> exit
ATM config>
```

ATM Virtual Interface Console Commands

This section summarizes the ATM virtual interface console commands. See “ATM-LLC Monitoring” on page 53-5 for additional information.

Chapter 53. Monitoring ATM

This chapter describes how to monitor the ATM interface. It includes the following sections:

Accessing the ATM Console Commands

Use the following procedure to access the ATM console commands. This process gives you access to an ATM's *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. (For more detail on this command, refer to Chapter 2, "The OPCON Process and Commands" on page 2-1.) For example:

   ```
   * talk 5
   +
   ```

   The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter configuration, press **Return** again.

2. Enter **interface** at the + prompt to display a list of configured interfaces.

3. Record the interface numbers.

4. Enter **network** followed by the number of the ATM interface.

   ```
     + network 5
   ATM+
   ```

   The ATM monitoring prompt (ATM+) is displayed.

ATM Console Commands

This section summarizes the ATM console commands for monitoring ATM interfaces. Enter the commands at the ATM+ prompt.

Table 53-1. ATM Console Command Summary

Command	Function
? (Help)	Lists all of the ATM configuration commands, or lists the options associated with specific commands.
INTERFACE	Displays the ATM Interface+ prompt from which you can monitor the ATM Interface, as described in "ATM Interface Console Commands (ATM INTERFACE+ Prompt)" on page 53-2.
ATM-LLC	Displays the ATM LLC+ prompt from which you can monitor endpoints, a set of user clients, and a set of ATM channels.
Exit	Exits the ATM Monitoring process and returns to the + prompt.

Interface

Displays the ATM Interface+ prompt, described in "ATM Interface Console Commands (ATM INTERFACE+ Prompt)."

ATM-LLC

Displays the ATM-LLC+ prompt, described in "ATM-LLC Monitoring" on page 53-5.

Exit

Returns you to the + prompt.

ATM Interface Console Commands (ATM INTERFACE+ Prompt)

This section summarizes and then explains the commands for monitoring a specific ATM interface.

Enter the commands at the ATM INTERFACE+ prompt.

Table 53-2. ATM INTERFACE Console Command Summary

Command	Function
? (Help)	List all the ATM Interface console commands, or list the options associated with specific commands.
List	Lists ATM addresses and VCCs.
Trace	Starts/Stops packet tracing on a specified VPI/VCI range. Trace can be viewed by ELS.
Wrap	Starts/Stops a loopback test on the VCC.
Exit	Returns to the ATM+ prompt.

List

Use the **list** command to list various categories of ATM data.

Syntax: list addresses
all
circuit
vccs
reserved-bandwidth

addresses

Lists the ATM addresses, along with a descriptive name, in use on the device.

Example: list addresses

```
ATM INTERFACE+ list addresses

                ATM Address                          Name
----------------------------------------  ------------------------------------
39999999999999000099990200000041347391804  LEC 1 'eth1'
39999999999999000099990200000041347391802  LES/BUS 'eth1'
```

all

Lists all of the following:

- Addresses
- Circuit statistics

- VCCs
- Reserved Bandwidth

circuit

Lists the statistics for a particular VCC by specifying the particular VCI-VPI pair. You can also specify the circuit on the command line; for example: list circuit 33.

Example: `list circuit`

```
ATM INTERFACE+ list circuit
VPI [0]?
VCI [32]?33

     Frames transmitted   =         2  Bytes transmitted   =       216
     Frames received      =         2  Bytes received      =       216
```

vccs

Lists all the VCCs established by the router. The VCCs may be permanent (PVC) or switched (SVC), point-to-point or point-to-multipoint, and each is identified by a unique VPI/VCI. The trace command uses the VPI/VCI value for a VCC to perform packet tracing over a particular VCC.

Example: `list vccs`

```
ATM Interface+ list vccs
                        VCCs
 VPI  VCI   Hndl   Type    FrmXmt     FrmRcv     ByteXmt    ByteRcv
 --- ----- ----- ------- ---------- ---------- ---------- ----------
   0   142    17    P-MP          0          0          0          0
 Name = BUS Mcast Fwd on 'eth1'

   0   143    19    P-MP          0          0          0          0
 Name = LEC 1 (LECID 0001) Mcast Fwd 'eth1'

   0   138    13  B0-139          1          0         62          0
 Name = LEC 1 (LECID 0001) Mcast Send 'eth1'

   0   139    16  B0-138          0          1          0         62
 Name = BUS Mcast Send LECID 0001 on 'eth1'

   0   134     9    P-MP          0          0          0          0
 Name = LES Cntrl Dist on 'eth1'

   0   135    11    P-MP          0          0          0          0
 Name = LEC 1 (LECID 0001) Cntrl Dist 'eth1'

   0   130     5     P-P          2          2        216        216
 Name = LEC 1 (LECID 0001) Cntrl Dir 'eth1'

   0   131     7     P-P          2          2        216        216
 Name = LES Cntrl Dir LECID 0001 on 'eth1'

   0     5     1    SAAL       2592       2592      27380      27036
 Name = SAAL

   0    16     2    ILMI        545        544      23646      35030
 Name = ILMI

VCC Totals: 4 point-to-point, 4 point-to-multipoint
ATM Interface+
```

P-P	point to point VCC
P-MP	point to multipoint VCC
ILMI	Interim Local Management Interface VCC
SAAL	signalling VCC
Bx-y	Internally bound VCC to VPI x, VCI y

Sx-y Internally spliced VCC to VPI x, VCI y

reserved-bandwidth

Lists the reserved bandwidth on the ATM Interface.

Example: `reserved-bandwidth`

```
ATM INTERFACE+ list reserved-bandwidth
Line Rate                    : 155000 Kbps
Peak Reserved Bandwidth      : None
Sustained Reserved Bandwidth : None
```

Trace

Use the **trace** command activate packet tracing over a specified range of VPI/VCI values. You can view trace data by using ELS as described in “View” on page 9-19.

Syntax: trace list
on
off

list

Displays the current packet tracing options on the ATM interface.

Example: `trace list`

```
ATM Interface+ trace
on | off | list []? list
Packet trace is ON
Range of VPIs to be traced:       0 -       0
Range of VCIs to be traced:      32 -      39
```

on

Starts packet tracing on all active VCCs within the specified VPI/VCI range.

Example: `trace on`

```
ATM Interface+ trace on
beginning of VPI range [0]?
end of VPI range [0]?
beginning of VCI range [32]?
end of VCI range [65535]? 39
```

off

Stops packet tracing on all VCCs.

Example: `trace off`

```
ATM Interface+ trace off
ATM Interface+ trace list
Packet trace is OFF
```

Wrap

Use the **wrap** command to perform a loopback data test on the ATM interface of the adapter. Wrap can be issued on a per VC basis by specifying VPI-VCI pairs. Data is looped back internally.

You can selectively start a wrap, stop a wrap, or display the current wrap settings.

If you stop or display a wrap, the following statistics will be displayed:

- Wrap transmits

- Wrap receives
- Wrap transmit errors
- Wrap receive errors
- Wrap receive timeouts

For display, the current wrap statistics are displayed.

For stop, the final wrap statistics are displayed.

Syntax: wrap display
start
stop

display
Displays the current wrap settings.

start
Starts the wrap procedure and specifies the VPI-VCI length of pattern and the pattern itself.

Example: `wrap start`

```
ATM Interface+ wrap start
VPI [0]?
VCI [32]?
wrap pattern length [32]?
Enter 32-byte wrap pattern: [ABCDEFGHIJKLMNOPQRSTUVWXYZ123456]?
```

stop
Stops the wrap procedure and displays final wrap statistics.

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

ATM-LLC Monitoring

This section explains the commands for monitoring ATM LLC multiplexing.

Enter the commands at the `ATM-LLC+` prompt.

Table 53-3. ATM LLC Configuration Command Summary

Command	Function
? (Help)	List all the ATM LLC configuration commands, or list the options associated with specific commands.
List	Lists various options
Exit	Returns to the `ATM+` prompt.

List

Use the **list** command to list various categories of ATM LLC monitoring data.

Syntax: list endpoints
channels

endpoints

Lists the ATM addresses in use by protocols using the ATM-LLC multiplexing function on the device. The endpoint is displayed as the End System Identifier and the Selector.

Example: `list endpoints`

```
ATM-LLC+ list endpoints
```

channels

Lists the channels in use by protocols using the ATM-LLC multiplexing function on the device.

Example: `list channels`

```
ATM-LLC+ list channels
```

Exit

Use the **exit** command to exit the **ATM-LLC+** command prompt.

Syntax: exit

Chapter 54. Using and Configuring LAN Emulation Clients

This chapter describes LAN Emulation Clients (LECs). It includes the following sections:

LAN Emulation Client Overview

On the router, LECs serve the purpose of "ports" or "interfaces" on traditional routers and bridges. The router bridges and routes traffic between ports by receiving and transmitting traffic through its LECs.

LEC has two prompt levels:

1. `LE Client Config>` lets you enter commands that control the environment of all your LECs. The commands for this prompt level are described in "Configuring LAN Emulation Clients (LE Client Config>)"
2. One of the commands, **config**, gets you to another prompt level, `LEC Config>`, at which you can enter commands to configure a specific LEC.

An explanation of commands for LAN Emulation Clients follows.

Configuring LAN Emulation Clients (LE Client Config>)

This section summarizes and explains the commands for configuring and using the set of LE Clients on a particular ATM interface.

To get to the `LE Client Config>` prompt, enter **le-c** at the ATM Config> prompt as described in "ATM Configuration Commands" on page 52-3.

Enter the commands at the `LE Client Config>` prompt under the `ATM Config>` prompt, as described in "ATM Configuration Commands" on page 52-3.

Table 54-1. LAN EMULATION Client Configuration Commands Summary

Command	Function
? (Help)	Lists the console commands or list the actions associated with specific commands
Add	Adds a LEC for the following types of ATM Forum-compliant Emulated LANs architectures: • Ethernet • Token Ring
Config	Gets you to the `LEC Config>` prompt, from which you can configure a specific LAN Emulation Client.
List	Lists the LEC.
Remove	Removes a LEC.
Exit	Returns to the previous prompt.

Help

Use the **? (help)** command to list the commands that are available from the current prompt level. You also can enter a ? after a specific command name to list its options.

Syntax: ?

Example: ?

```
ADD
CONFIG
LIST
REMOVE
EXIT
```

Add

Use the **add** command to add a LEC for a Token-Ring or Ethernet emulated LAN.

Syntax: add Ethernet
Token Ring

token-ring
Token-ring emulated LAN

Example: add token ring

```
LE Client Config> add token-ring
Added Emulated LAN as interface 3
```

ethernet
Ethernet emulated LAN

Example: add ethernet

```
LE Client Config> add ethernet
Added Emulated LAN as interface 2
```

Config

Use the **config** command to get you to the LEC Config> prompt, from which you can configure the details of a specific LAN Emulation Client.

Syntax: config interface#

interface#
An integer number assigned by the router when the LEC was added to the configuration. Use the **list** command to determine the interface number assigned to the LEC.

Example: config

```
LE Client Config> config 3
ATM LAN Emulation Client configuration
```

List

Use the **list** command to list the LAN emulation clients.

Syntax: list

Example: list

```
LE Client Config> list
                  ATM Forum Compliant Emulated LANs
-----------------------------------------------------------
  Physical ATM interface number = 0
  LEC interface number = 1
  Emulated LAN type    = Token Ring Forum Compliant
  Emulated LAN name    =
```

Remove

Use the **remove** command to remove a LEC. You must specify the interface number that was assigned when the LEC was added to the configuration. Use the **list** command to determine the interface number assigned to the LEC.

Syntax: remove interface#

interface#
An integer number assigned by the router.

Exit

Use the **exit** command to return to the previous prompt.

Configuring an ATM Forum-Compliant LE Client

This section explains the commands for configuring an ATM Forum-compliant LAN Emulation Client. Enter the appropriate commands at either the `Ethernet-Forum LE Client Config>` prompt or the `Token-Ring-Forum LE Client Config>` prompt. Commands in the following table apply to both types of LECs except where indicated.

Enter the commands at the `LEC Config>` prompt after entering the **config** command at the `LE Client Config>` prompt.

Table 54-2. LAN Emulation Client Configuration Commands Summary

Command	Function
? (Help)	Lists the console commands or lists the actions associated with a particular command.
ARP-Configuration	Allows you to configure the LE-ARP configuration for the ATM Forum-compliant client
Frame	Sets the NetWare IPX encapsulation type.
RIF-Timer	Sets the maximum amount of time that information in the RIF is maintained before it is refreshed. Applies only to Token-Ring LECs.
Source-routing	Used to enable or disable source-route bridging. Applies only to Token-Ring LECs.
IP-Encapsulation	Sets the IP encapsulation as Ethernet (type X'0800') or IEEE (802.3 with SNAP). Applies only to Ethernet LECs.
List	Lists the LAN Emulation Client configuration.
QOS-Configuration	Gets you to the `elan-x LEC QoS Config>` prompt from which you can configure Quality of Service as described in “LE Client QoS Configuration Commands” on page 50-7.
Set	Sets the LAN Emulation Client parameters.
Exit	Returns to the previous prompt.

Help

Use the **? (help)** command to list the commands that are available from the current prompt level. You also can enter a ? after a specific command name to list its options.

Syntax: ?

Example: ?

```
ARP-CONFIGURATION
FRAME
LIST
RIF-TIMER
SET
SOURCE-ROUTING
QOS-CONFIGURATION
EXIT
```

ARP Configuration

Use the **ARP-configuration** command to configure the static LE-ARP entries for the ATM forum-compliant LAN Emulation Client.

Syntax: ARP-configuration

Example:

```
Token Ring Forum Compliant LEC Config> arp-configuration
ATM LAN Emulation Clients ARP configuration
```

Table 54-3. ATM LAN Emulation Client ARP Configuration Commands Summary

Command	Function
? (Help)	Lists the console commands or lists the actions associated with a particular command.
Add	Adds an LE-ARP cache entry using a MAC or route descriptor ARP
Config	Sets cache entry QOS parameter values.
List	Lists configured ARP cache entries.
Remove	Removes an ARP cache entry.
Exit	Returns to the previous prompt.

Add

Use the Add command to add an ARP cache entry using the MAC address or a route descriptor.

MAC addresses, and route descriptors are entered as strings of hexadecimal characters with or without optional separator characters between bytes. Valid separator characters are dashes (–), periods (.), or colons (:).

Add mac
route descriptor

Config

Use this command to provide configuration parameter information for the ARP you are configuring. See Chapter 50, “Using and Configuring Quality of Service (QoS)” on page 50-1 for additional information about ARP configuration parameters.

Use the **Config** command to configure the permanent ARP cache entry QOS parameters for the ATM forum-specific LAN Emulation Client.

Syntax: Config *arp-entry-number*

Example:

```
ARP config for LEC> config
ARP entry number [1]
Configure LEC ARP entry
```

Table 54-4. ATM LAN Emulation Client ARP Config Commands Summary

Command	Function
? (Help)	Lists the console commands or lists the actions associated with a particular command.
Set	Sets QOS parameter values.
Exit	Returns to the previous prompt.

Help: Use the **? (help)** command to list the commands that are available from the current prompt level. You also can enter a ? after a specific command name to list its options.

Syntax: ?

Example: ?

```
SET
EXIT
```

Set: Use the **Set** command to configure the permanent ARP cache entry QOS parameters for the ATM forum-specific LAN Emulation Client.

Syntax: Set . . .

Example:

```
ARP entry 'identifier' config> set ?
 MAX-RESERVED-bANDWIDTH
 TRAFFIC-TYPE
 PEAK-CELL-RATE
 SUSTAINED-CELL-RATE
 QOS-CLASS
 MAX-BURST-SIZE
```

See Chapter 50, “Using and Configuring Quality of Service (QoS)” on page 50-1 for information about the following QoS parameters.

Set max-reserved-bandwidth
traffic-type
peak-cell-rate
sustained-cell-rate
qos-class

max-burst-size

Exit: Returns to the previous prompt.

List

Use the **List** command to display information about ARP configuration.

Remove

Use this command to remove an configured MAC address or Route Descriptor LE-ARP entry.

Select the ARP entry number to be removed from the list provided.

Remove *arp-entry-number*

Exit

Use this command to exit the ARP configuration and return to the LE-client configuration.

Frame

Use the **frame** command to set the NetWare IPX encapsulation type. The command options differ depending on the type of LEC (Token-Ring or Ethernet). For Token-Ring LECs, enter one of the following:

Option	Description	Syntax
Token-Ring using MSB	Uses the standard 802.2 IPX header with the noncanonical Token-Ring address bit ordering (MSB).	frame token-ring msb
Token-Ring using LSB	Uses the 802.2 IPX header with the canonical address bit ordering (LSB).	frame token-ring lsb
Token-Ring with 802.2 SNAP using MSB	Uses the 802.2 format with a SNAP header and noncanonical address bit ordering. This encapsulation is used primarily in bridging environments.	frame token-ring_snap msb
Token-Ring with 802.2 SNAP using LSB	Uses the 802.2 format with a SNAP header and canonical address bit ordering.	frame token-ring_snap lsb
Ethernet 2.0	Uses Ethernet version 2.0 protocol 81-37.	frame ethernet_II
Ethernet 802.2	Uses Ethernet 802.3 with 802.2 SA E0	frame ethernet_8022
Ethernet 802.3	Uses Ethernet 802.3 without any 802.2 header	frame ethernet_802.3
Ethernet SNAP	Uses 802.3, 802.2 with SNAP PID 00-00-00-81-37	frame ethernet_SNAP

Syntax: frame *ipx-encapsulation type*

Note: The frame command cannot be used in the network configuration process to set the IPX encapsulation unless the interface has been configured with IPX.

The IPX encapsulation can also be set in the IPX configuration environment. Refer to the Protocol Configuration and Monitoring Reference chapter on Configuring IPX for details.

Example: `frame token_ring msb`

RIF-Timer (for Token-Ring Forum-compliant LEC only)

Use the **RIF-Timer** command to set the maximum amount of time that information in the RIF is maintained before it is refreshed. Range is 0 to 4096. The default is 120.

Syntax: rif-timer

Example: `rif-timer 100`

Source-routing (for Token-Ring Forum-compliant LEC only)

Use the **source-routing** command to enable or disable end station source-routing. Source routing is the process by which end stations determine the source route to use to cross source routing bridges. Source routing allows the IP, IPX, and AppleTalk Phase 2 protocols to reach nodes on the other side of the source route bridge.

This function of the device is not changed whether source routing is enabled or disabled. The default setting is enabled.

Some stations cannot properly receive frames with Source Routing RIF on them. This is especially common among NetWare drivers. Disabling source routing in this situation will allow you to communicate with these stations.

Source routing should be enabled only if there are source-routing bridges on this ring through which you want to bridge IP, IPX, and AppleTalk Phase 2 packets. Source routing must also be enabled so that LLC test response messages can be returned.

Syntax: source-routing enable
disable

Example: `source-routing disable`

IP-Encapsulation (for Ethernet ATM Forum-compliant LEC only)

Use the **IP-encapsulation** command to select Ethernet (Ethernet type X'0800') or IEEE 802.3 (Ethernet 802.3 with SNAP). Specify either type **E**thernet or **I**EEE-803.3.

Syntax: IP-encapsulation *type*

Example: `IP-encapsulation E`

List

Use the **list** command to list the LE client configuration.

Syntax: list

Example: `list`

QoS

Use the **qos-configuration** command to get you to the `LEC QoS Config>` prompt from which you can configure Quality of Service as described in "LE Client QoS Configuration Commands" on page 50-7.

Syntax: qos-configuration

Set

Use the **set** command to set LE Client parameters.

Syntax: set
- arp-aging-time
- arp-cache-size
- arp-queue-depth
- arp-response-time
- auto-config
- best-effort-peakrate
- config-retries
- conn-completion-time
- control-timeout
- elan-name
- esi-address
- flush-timeout
- forward-delay
- frame-size
- lecs-atm-address
- les-atm-address
- mac-address
- multicast-send-avg
- multicast-send-peak
- multicast-send-type
- path-switch-delay
- retry-count
- selector
- trace
- unknown-count
- unknown-time
- vcc-timeout

arp-aging-time

Sets ARP aging time. This is the maximum time that a LEC will maintain an entry in its LE_ARP cache in the absence of a verification of that relationship. A larger aging time may result in a faster session setup time, but may also use more memory and reacts slower to changes in network configuration.

Valid Values: An integer number of seconds in the range of 10 to 300.

Default Value: 300

Example: `set arp-aging-time`

```
LEC Config> set aging-time 200
```

arp-cache-size

Sets the number of entries in the ARP cache. The size of the ARP cache limits the number of simultaneous data direct VCCs. Larger ARP caches require more memory, but permit the client to simultaneously converse with a larger number of destinations.

Valid Values: An integer number in the range of 10 to 1024.
Default Value: 1024

Example: `set arp-cache-size`

```
LEC Config> set arp-cache-size 10
```

arp-queue-depth

Sets the maximum number of queued frames per ARP cache entry. The LEC enqueues frames when switching the data path from the Multicast Send VCC to a Data Direct VCC. Frames passed to the LEC for transmission will be discarded if the queue is full. A larger queue requires more memory, but results in fewer discarded frames during the data path switch.

Valid Values: An integer number in the range of 0 to 10.
Default Value: 5

Example: `set arp-queue-depth`

```
LEC Config> set arp-queue-depth 10
```

arp-response-time

Sets expected ARP response time. This value controls how frequently an unanswered LE ARP request is retried. Larger values result in fewer LE ARPs, which causes less traffic and possibly increase the amount of time before a Data Direct VCC is established.

Valid Values: An integer number of seconds in the range of 1 to 30.
Default Value: 1 second

Example: `set arp-response-time`

```
LEC Config> set arp-response-time 20
```

auto-config

Specifies whether this LEC uses LECS auto-config mode. Specify YES or NO. The LEC may contact the LECS to obtain the address of its LES and various other configuration parameters.

Valid Values: If YES, then you do not have to configure the ATM address of the LES.

If NO, then you *must* configure the ATM address of the LES using the **set les-atm-address** command as described on page 54-12.

Default Value: NO

Example: `set les auto-config`

```
LEC Config> set les auto-config yes
```

best-effort-peakrate

Sets the Best Effort Peak Rate. Used when establishing best effort multicast send connections.

The maximum peak rate depends on the maximum data rate of the ATM device.

Specify an integer from 1 to the maximum peak rate in Kbps (the definition is the maximum data rate) as follows:

- If ATM maximum data rate is 25 Mbps, the maximum peak rate is 25,000 Kbps.
- If ATM maximum data rate is 155 Mbps, the maximum peak rate is 155,000 Kbps.

Valid Values: An integer number in the range of 1 - device maximum data rate.
Default Value: 155000

Example: `set best-effort-peakrate`

```
LEC Config> set best-effort-peakrate 24000
```

config-retries

Sets the number of configuration retries.

If the client is unable to connect to the LES address given to it in a configure response, the client will send another configure request to the LECS before releasing the configuration direct VCC. This value controls how many retries are made before the client releases the configuration direct VCC. When this limit is reached, the client will sleep for a period of time (not exceeding 30 seconds) and restart the configuration phase. Interoperability problems may arise if this value is set to zero.

Valid Values: An integer number in the range of 0 to 5.
Default Value: 3

Example: `set config-retries`

```
LEC Config> set config-retries 4
```

connection-completion-time

Sets the connection completion time. This is the time interval in which data or a READY_IND message is expected from a calling party.

When a Data Direct VCC is established to the client, the LEC expects data or a READY_IND message within this time period. The LEC will not transmit frames over a Data Direct VCC established to it until receiving data or a READY_IND. This parameter value controls the amount of time which passes before the LEC issues a READY QUERY (in hopes of receiving a READY_IND). Smaller values lead to faster response times, but also to unnecessary transmissions.

Valid Values: An integer number of seconds in the range of 1 to 10.
Default Value: 4 seconds

Example: `set connection-completion-time`

```
LEC Config> set connection-completion-time 5
```

control-timeout

Sets the control timeout. This is the timeout period used for timing out most request/response control frame interactions. Smaller timeout values result in more traffic and may result in faster response times.

Valid Values: An integer number of seconds in the range of 10 to 300.
Default Value: 30 seconds

Example: `set control-timeout`

```
LEC Config> set control-timeout 100
```

elan-name

Specifies name of the ELAN that the LEC wishes to join. This is the ELAN name sent to the LECS in the configure request (if the LEC autoconfigures) or to the LES in the join request. The LECS or LES may return a different ELAN name in the response.

Valid Values: Any character string length of 0 - 32 bytes.
Default Value: Blank

Note: A blank name (0 length string) is valid.

Example: `set elan-name`

```
LEC Config> set elan-name FUZZY
```

esi-address

Sets the ESI portion of the LEC's ATM address.

Specify the ESI portion (octets 13 through 19) of the LEC's ATM address. The ESI and selector combination of the LEC must be unique among all LAN emulation components on the device.

Valid Values: Any 12 hexadecimal digits.
Default Value: none

Example: `set esi`

```
Select ESI
   (1) Use burned in ESI
   (2) 11.22.33.44.55.66

Enter selection [1]?
```

flush-timeout

Sets the flush timeout. This is the time limit to wait to receive the LE_FLUSH_RESPONSE after the LE_FLUSH_REQUEST has been sent before taking recovery action. This is the amount of time between flush requests when the LEC is attempting to switch to a Data Direct connection.

When switching from the multicast send to a data direct data path, the client sends a flush request over the multicast send VCC. Until a flush response is received, or until the path switch delay expires, frames are queued for the destination. A smaller timeout generates more flush traffic, but may result in fewer discarded data frames.

Valid Values: An integer number of seconds in the range of 1 to 4.
Default Value: 4 seconds

Example: `set flush-timeout`

```
LEC Config> set flush-timeout 3
```

forward-delay

Sets the forward delay. Entries in the LE ARP cache must be periodically reverified. The forward delay time is the maximum amount of time a remote entry may remain in the cache during a network topology change. Larger aging times may result in stale (invalid) entries, but also cause less reverification traffic.

Valid Values: An integer number of seconds in the range of 4 to 30.

Default Value: 15 seconds

Example: `set forward-delay`

```
LEC Config> set forward-delay 10
```

frame-size

Sets the frame size.

The value specified for frame-size must be equal to or less than the value specified for ATM max-frame using the ATM INTERFACE> **set max-frame** command as described on page 52-7.

Valid Values: 1516

4544

9234

18190

Default Value: If the ELAN type is token ring, the default is 4544. If the ELAN type is Ethernet, the default is 1516.

Example: `set frame-size`

```
LEC Config> set frame-size 4544
```

lecs-atm-address

Specifies the ATM address of the LECS.

If the client is set to auto configure, it attempts to connect to a LECS. If it is unable to connect to a LECS, then it may try another LECS ATM address. The LECS ATM addresses that are tried, in order, are:

1. This configured LECS address
2. Any LECS address obtained through ILMI
3. The well-known LECS address defined by the ATM Forum.

No default is provided.

Note: This command should be entered on one command line. It is shown here on two lines because of spacing.

Example: `set lecs-atm-address`

```
LEC Config> set lecs-atm-address
39.84.0F.00.00.00.00.00.00.00.00.01.10.00.5A.00.DE.AD.01
```

les-atm-address

Sets the LES ATM address. This command may be optional or required depending upon the setting of lecs-auto-config as described in the **set lecs-auto-config** command on page 54-9.

- If lecs-auto-config is YES, the les-atm-address is not configurable.
- If lecs-auto-config is NO, then the les-atm-address is required.

Specify the ATM address of the LES. No default is provided.

Note: This command should be entered on one command line. It is shown here on two lines because of spacing.

Example: `set les-atm-address`

```
LEC Config> set les-atm-address
      39.84.0F.00.00.00.00.00.00.00.00.01.10.00.5A.00.DE.AD.02
```

mac-address

Sets the MAC address for this LE client. You *may* specify that the client use the burned-in MAC address of the ATM interface, or you may specify a different MAC address. If you have two clients that are bridged together, they should use different MAC addresses.

This MAC address is registered with the LES when the client joins the ELAN.

Valid Values: Any valid MAC address.
Default Value: none

Example: `set mac-address`

```
LEC Config> set mac-address FF.FF.FF.FF.FF.01
```

multicast-send-avg

Sets the multicast send VCC average rate in Kbps. Used by the LEC for reserving bandwidth on the VCC to the BUS. It specifies the forward and backward sustained cell rate used when setting up a reserved bandwidth multicast send VCC.

This parameter is only applicable when the multicast-send-type is reserved bandwidth. If multicast-send-avg equals multicast-send-peak, then a constant bit rate (CBR) multicast send is signalled. Otherwise, a variable bit rate (VBR) multicast send is signalled. Multicast-send-avg must at least equal multicast-send peak.

A reserved bandwidth multicast send VCC may improve data transfer rates in congested networks, but reserving bandwidth and not using it wastes network resources.

When the multicast-send-type is reserved, then multicast-send-avg and multicast-send-peak must be specified.

Example: `set multicast-send-avg`

```
LEC Config> set multicast-send-avg 4000
```

multicast-send-peak

Sets the multicast send peak rate in Kbps. Used by LEC for reserving bandwidth on the VCC to the BUS. It specifies the forward and backward peak cell rate used when establishing a reserved bandwidth multicast send VCC.

This parameter is only applicable when the multicast-send-type is reserved bandwidth. If multicast-send-avg equals multicast-send-peak, then a constant bit rate (CBR) multicast send is signalled. Otherwise, a variable bit rate (VBR) multicast send is signalled. Multicast-send-avg must at least equal multicast-send peak.

A reserved bandwidth multicast send VCC may improve data transfer rates in congested networks, but reserving bandwidth and not using it wastes network resources.

When the multicast-send-type is reserved, then multicast-send-avg and multicast-send-peak must be specified.

Example: `set multicast-send-peak`

```
LEC Config> set multicast-send-peak 155
```

multicast-send-type

Sets the multicast send type. Specifies the method used by the LEC when establishing the multicast send VCC.

If multicast-send-avg equals multicast-send-peak, then a constant bit rate (CBR) multicast send is signalled. Otherwise, a variable bit rate (VBR) multicast send is signalled. Multicast-send-avg must at least equal multicast-send peak.

A reserved bandwidth multicast send VCC may improve data transfer rates in congested networks, but reserving bandwidth and not using it wastes network resources.

When the multicast-send-type is reserved, then multicast-send-no and multicast-send-peak must be specified.

Valid Values: Best Effort or Reserved
Default Value: Best Effort

Example: `set multicast-send-type`

```
LEC Config> set multicast-send-type best-effort
```

path-switch-delay

Sets the path switch delay.

The LEC must ensure that all frames sent through the BUS to a destination have arrived at the destination before it can start using a Data Direct VCC. This is accomplished using the flush protocol, or by waiting path-switch-delay seconds after sending the last packet to the BUS. Smaller values improve performance, but may result in out-of-order packets in a heavily congested network.

Valid Values: An integer number of seconds in the range of 1 to 8.
Default Value: 6

Example: `set path-switch-delay`

```
LEC Config> set path-switch-delay 5
```

retry-count

Sets the retry count. This is maximum number of times that the LEC retries an LE_ARP_REQUEST for a specific frame's LAN destination. f no ARP response is received after the specified number of retries, then the entry is purged from the LE ARP cache.

Valid Values: An integer number in the range of 0 to 2.
Default Value: 1

Example: `set retry-count`

```
LEC Config> set retry-count 2
```

selector

Specifies the selector portion of the client's ATM address. The combination of ESI and selector must be unique among all LANE components on the MSS Server. By default, a unique selector is selected for the configured ESI.

Valid Values: Any octet, in hexadecimal, that is not in use by another LANE component with the same ESI.

Example: `set selector`

```
LEC Config> set selector 01
```

trace

Enables tracing for the LEC. To perform packet tracing, three steps are required:

1. Enable packet tracing system (under ELS)
2. Enable tracing on the LEC subsystem (under ELS)
3. Enable packet tracing on the desired LECs (using this command).

Valid Values: Enable or Disable
Default Value: Disable

Example:

```
Token Ring LEC config>set trace
Trace packets on the LEC? [No]?yes
```

unknown-count

Sets the unknown frame count. This is the maximum number of frames for a specific unicast MAC address or route descriptor that may be sent to the BUS within the time specified by the unknown-time parameter. Larger values decrease the number of discarded frames while increasing the load on the BUS.

Valid Values: An integer number of frames in the range of 1 to 255.
Default Value: 10

Example: `set unknown-count`

```
LEC Config> set unknown-count
```

unknown-time

Sets the unknown frame time. This is the time interval during which the maximum number of frames for a specific unicast MAC address or route descriptor (specified by the unknown-count parameter) may be sent to the BUS. Larger values increase the number of discarded frames while decreasing the load on the BUS.

Valid Values: An integer number of seconds in the range of 1 to 60.
Default Value: 1

Example: `set unknown-time`

```
LEC Config> set unknown-time 5
```

vcc-timeout

Sets the VCC timeout. Data direct VCCs over which no traffic has been sent for this period of time should be released.

Specify an integer number in the range 0 to 31536000 seconds (1 year).

Default: 1200

Note: This parameter is meaningful only for SVC connections.

Example: `set vcc-timeout`

```
LEC Config> set vcc-timeout 1000
```

Exit

Use the **exit** command to return to the `LE Client Config>` prompt.

Chapter 55. Monitoring LAN Emulation Clients

This chapter describes how to monitor LAN Emulation Clients (LECs). It includes the following sections:

- "Accessing the LEC Console Environment"
- "LEC Console Commands"

Accessing the LEC Console Environment

Use the following procedure to access the LEC console commands. This process gives you access to the LEC *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. (For more detail on this command, refer to Chapter 2, "The OPCON Process and Commands" on page 2-1.) For example:

   ```
   * talk 5
   +
   ```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the console. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the + prompt, enter the **network** command to display the network interface numbers for which the router is currently configured, and enter the *interface number* for the LEC you wish to monitor. For example:

   Example: `network`

   ```
   + network

   1  : ATM Ethernet LAN Emulation: ETH
   2  : IP Protocol Network
   3  : Bridge Application
   5  : CHARM ATM Adapter
   Network number [0]? 1
   LEC+
   ```

   The LEC monitoring prompt (`LEC+`), is displayed.

   If you know the interface number of the LEC you wish to monitor, enter the **network** command followed by the *interface number* of the LEC.

   ```
   + network 1
   LEC+
   ```

LEC Console Commands

This section summarizes and then explains the LEC console commands. You can access LEC console commands at the `LEC+` prompt. Table 55-1 on page 55-2 shows the commands.

Table 55-1. LE Config Console Command Summary

Command	Function
? (Help)	Lists the LEC console commands or lists the options associated with specific commands.
List	Lists: • LEC Address Resolution Table (ARP) • LEC configuration • Data Direct VCC information • LEC statistics.
MIB	Displays LEC MIB objects including: • LEC MIB Configuration Table • LEC MAC ARP Table • LEC Route Descriptor Table • LEC MIB Server VCC Tables • LEC MIB Statistics Table • LEC MIB Status Table
QoS	Gets you to the `LEC x QoS+` prompt from which you can monitor Quality of Service as described in “Quality of Service Console Commands” on page 51-1.
Trace	Sets packet tracing on or off
Exit	Exits the LEC console process.

? (Help)

Use the **? (help)** command to list the commands that are available from the current prompt level. You can also enter a ? after a specific command name to list its options.

Syntax: ?

Example: **`?`**

List

Use the **list** command to list the LEC Address Resolution Table (ART), list the LEC configuration, list Data Direct VCC information, or list LEC statistics.

Syntax: list arp-table
configuration
data-direct-vccs
statistics
vcc-table

arp

Lists the LEC Address Resolution Table (entries in the ARP cache).

Example: **`list arp`**

```
LEC+ list arp

        LEC Address Resolution (LE ARP Cache) Table

  Max Table Size          = 10         Total # of entries
  Free Table Entries      = 10         # of free entries.
  Current Mac Entries     = 0          MAC - ATM  entries
  Current RD Entries      = 0          RD - ATM entries
  Arp Aging Time          = 300        Time for an entry to be 'aged' out
  Verify Sweep Interval   = 60

                                Xmit  BUS   Arp
                           Conn Queue Frame Retry Aging
 MAC Address     Remote Handle Depth Count Count Timer  Destination ATM Ad
                                                                    dress
 ------------------------------------------------------------------------
 40.00.00.00.00.09 False 652   0     0     0     60     39.99.99.99.99.99.
                               99.00.00.99.99.30.02.40.00.00.00.00.09.81
```

Note: The Sweep Interval is always one-fifth of the ARP Aging Timer value.

Max Table Size
: The total number of entries available

Free Table Entries
: The number of free entries

Current MAC Entries

Current RD Entries
: Route Descriptor ATM entries

ARP Aging Time
: Time for an entry to be aged out

Verify Sweep Interval

MAC Address

Remote

Connection Handle

Queue Depth

Xmit Frame Count

BUS Retry Count

ARP Aging Timer

Destination ATM Address

configuration
: Lists the LEC configuration.

  For Ethernet:

  Example: `list config`

```
LEC+ list config

                ATM LEC Configuration

   ATM interface number             = 0
   LEC interface number             = 1
   LECS auto configuration          = No

   C1: Primary ATM address
          ESI address               = Use burned in addr
          Selector byte             = 3
   C2: Emulated LAN type            = Ethernet
   C3: Maximum frame size           = 1516
   C5: Emulated LAN name            =
   C6: LE Client MAC address        = Use burned in addr
   C7: Control timeout              = 120
   C9: LE Server ATM address        = 39.84.0F.00.00.00.00.00.00.00.00.00.01.10.00.5A.00.DE.AD.02
   C10: Maximum unknown count        = 1
   C11: Maximum unknown time         = 1
   C12: VCC timeout period           = 1200
   C13: Maximum retry count          = 1
   C17: Aging time                   = 300
   C18: Forward delay time           = 15
   C20: LE ARP response time         = 1
   C21: Flush timeout                = 4
   C22: Path switch delay            = 6
   C24: Multicast send VCC type      = Best-Effort
   C25: Multicast send VCC avg rate  = 25000000
   C26: Multicast send VCC peak rate = 25000000


   C28: Connection completion timer  = 4

   LE ARP queue depth               = 10
   LE ARP cache size                = 10
   Best effort peak rate            = 25000

   Maximum config retries           = 3
   Packet Trace                     = No
   IP Encapsulation                 = Ether
   No IPX Interface Configuration
```

For Token Ring:

Example: `list config`

```
LEC+list config

                ATM LEC Configuration

  ATM interface number            = 0
  LEC interface number            = 1
  LECS auto configuration         = No

  C1: Primary ATM address
        ESI address               = Use burned in addr
        Selector byte             = 2
  C2: Emulated LAN type           = Token Ring
  C3: Maximum frame size          = 4544
  C5: Emulated LAN name           =
  C6: LE Client MAC address       = 10.00.5A.11.11.11
  C7: Control timeout             = 120
  C9: LE Server ATM address       = 39.84.0F.00.00.00.00.00.00.00.00.01.10.00.5A.00.DE.AD.02
 C10: Maximum unknown count       = 1
 C11: Maximum unknown time        = 1
 C12: VCC timeout period          = 1200
 C13: Maximum retry count         = 1
 C17: Aging time                  = 300
 C18: Forward delay time          = 15
 C20: LE ARP response time        = 1
 C21: Flush timeout               = 4
 C22: Path switch delay           = 6
 C24: Multicast send VCC type     = Best-Effort
 C25: Multicast send VCC avg rate = 25000
 C26: Multicast send VCC peak rate = 25000


 C28: Connection completion timer  = 4

  LE ARP queue depth               = 10
  LE ARP cache size                = 36920
  Best effort peak rate            = 25000

  Maximum config retries           = 999
  Packet trace                     = Yes
  RIF Aging Timer                  = 120
  Source Routing                   = Enabled
  IPX interface configuration record missing
```

See “Set” on page 54-8 for a definition of the parameters shown in the above examples.

data

Lists the LEC Data Direct VCC information.

Example: `list data`

```
LEC+ list data

         LEC Data Direct VCC Table

   Max Table Size     = 1019    Max no of SVC connections
   Current Size       = 0       Currently used
   Inactivity Timeout = 1200    No Data Xfer Timeout before connection is
                                closed (seconds)
   Sweep Interval     = 60
      Conn              Inactive  User
      Handle  VPI  VCI   Timer    Count  Destination ATM Address
    ------------------------------------------------------------------------

       652     0  7241    300       1    39.99.99.99.99.99.99.00.00.99.99.30.02.
                                         40.00.00.00.00.09.81


    ------------------------------------------------------------------------
```

statistics

Lists LEC statistics.

Example: `list statistics`

```
LEC+ list stat

         LEC Statistics

   In Octets.high     = 0       No of Bytes received
   In Octets.low      = 346
   In Discards        = 2       Packets discarded
   In Errors          = 0       Rx.Errors
   In Unknown Protos  = 0       Unknown protocols received
   Out Octets.high    = 0       No of Bytes xmitted.
   Out Octets.low     = 0
   Out Discards       = 0
   Out Errors         = 0       Tx.Errors
   In Frames          = 0
   Out Frames         = 0
   In Bytes           = 0
   Out Bytes          = 0
```

MIB

Use the **mib** command to display MIB objects.

Note: Some of this information may be displayed in a different format using the **list** command.

Syntax: mib config-table
mac-arp-table
rd-arp-table
server-vcc-table
statistics-table
status-table

config

Displays the LEC MIB Configuration Table.

Example: `list mib config`

```
LEC+ mib config

 lecConfigTable:
 lecConfigMode                = Manual
 lecConfigLanType             = 802.3 - Ethernet
 lecConfigMaxDataFrameSize    = 1516
 lecConfigLanName             =
 lecConfigLesAtmAddress       = 39.84.0F.00.00.00.00.00.00.11.23.24.24.24.24.55.66.77.88.99.00
 lecControlTimeout            = 120
 lecMaxUnknownFrameCount      = 1
 lecMaxUnknownFrameTime       = 0
 lecVccTimeoutPeriod          = 1200
 lecMaxRetryCount             = 1
 lecAgingTime                 = 300
 lecForwardDelayTime          = 15
 lecExpectedArpResponseTime   = 1
 lecFlushTimeout              = 4
 lecPathSwitchingDelay        = 6
 lecLocalSegmentId            = 0
 lecMulticastSendType         = 1
 lecMulticastSendAvgRate      = 25000000
 lecMulticastSendPeakRate     = 25000000


 lecConnectionCompleteTimer   = 4
```

lecConfigMode LEC config mode: AUTO or MANUAL. If AUTO, LEC Uses LECS to get the LES ATM address.

lecConfigLanType LAN type, either Ethernet or token-ring

lecConfigMaxDataFrameSize
Maximum frame size

lecConfigLanName ELAN Name

lecConfigLesAtmAddress
LE Server ATM address

lecControlTimeout Timeout for request/response control frame

lecMaxUnknownFrameCount
Maximum number of unknown frames

lecMaxUnknownFrameTime
Period in which LEC will send a maximum of MaxUnknownFrameCount frames to the BUS for a given unicast LAN Destination, and it must also initiate the address resolution protocol to resolve that LAN Destination.

lecVccTimeoutPeriod Inactivity timeout of SVC Data Direct VCCs

lecMaxRetryCount LE ARP retry count

lecAgingTime Life of unverified entry in the ARP table

lecForwardDelayTime

lecExpectedArpResponseTime
ARP Request/Response cycle time

lecFlushTimeout LE Flush Request/Flush Reply timeout period

lecPathSwitchingDelay

lecLocalSegmentId Segment ID of emulated LAN. Only for 802.5 clients

lecMulticastSendType
Signaling parameter used by LEC for multicast send VCC

lecMulticastSendAvgRate
Signaling parameter used by LEC for multicast send VCC

lecMulticastSendPeakRate
Signaling parameter used by LEC for multicast send VCC

lecConnectionCompleteTimer

mac
Displays the LEC MAC ARP Table

rd
Displays the LEC Route Descriptor Table

server

Displays the LEC MIB Server VCC Tables

Example: `mib server`

```
LEC+ mib server

lecServerVccTable:
   lecConfigDirectInterface      = 0
   lecConfigDirectVpi            = 0
   lecConfigDirectVci            = 0
   lecControlDirectInterface     = 1
   lecControlDirectVpi           = 0
   lecControlDirectVci           = 38
   lecControlDistributeInterface = 1
   lecControlDistributeVpi       = 0
   lecControlDistributeVci       = 37
   lecMulticastSendInterface     = 1
   lecMulticastSendVpi           = 0
   lecMulticastSendVci           = 34
   lecMulticastForwardInterface  = 1
   lecMulticastForwardVpi        = 0
   lecMulticastForwardVci        = 33
```

lecConfigDirectInterface	The interface associated with the Configuration Direct VCC
lecConfigDirectVpi	VPI which identifies the above VCC if it exists
lecConfigDirectVci	VCI which identifies the above VCC if it exists
lecControlDirectInterface	The interface associated with the Control Direct VCC
lecControlDirectVpi	VPI which identifies the above VCC if it exists
lecControlDirectVci	VCI which identifies the above VCC if it exists
lecControlDistributeInterface	The interface associated with the Control Distribute VCC
lecControlDistributeVpi	VPI which identifies the above VCC if it exists
lecControlDistributeVci	VCI which identifies the above VCC if it exists
lecMulticastSendInterface	The interface associated with the Multicast Send VCC
lecMulticastSendVpi	VPI which identifies the above VCC if it exists
lecMulticastSendVci	VCI which identifies the above VCC if it exists
lecMulticastForwardInterface	The interface associated with the Multicast Forward VCC
lecMulticastForwardVpi	VPI which identifies the above VCC if it exists
lecMulticastForwardVci	VCI which identifies the above VCC if it exists

statistics

Displays the LEC MIB Statistics Table.

Example: `mib statistics`

```
LEC+ mib statistics

lecStatisticsTable:
   lecArpRequestsOut             = 1
   lecArpRequestsIn              = 0
   lecArpRepliesOut              = 0
   lecArpRepliesIn               = 1
   lecControlFramesOut           = 2
   lecControlFramesIn            = 2
   lecSvcFailures                = 1
```

lecArpRequestsOut — No. of LE ARP requests sent by this LEC

lecArpRequestsIn — No. of LE ARP requests received by this LEC

lecArpRepliesOut — No. of LE ARP responses sent by this LEC

lecArpRepliesIn — No. of LE ARP responses received by this LEC

lecControlFramesOut — No. of Control Packets sent by this LEC

lecControlFramesIn — No. of Control Packets received by this LEC

lecSvcFailures — The total number of:

- Outgoing LAN Emulation SVCs which this client tried but failed, to open
- Incoming LAN Emulation SVCs which this client tried, but failed to establish
- Incoming LAN Emulation SVCs which this client rejected for protocol or security reasons

status

Lists MIB status.

Example: `mib status`

```
LEC+ mib status

lecStatusTable:
   lecPrimaryAtmAddress          = 39.84.0F.00.00.00
   Client ATM address=           = 00.00.00.00.00.01.10.00.5A.00.DE.AD.03
   lecId                         = 1                    Assigned by LES
   lecInterfaceState             = Operational          State of the LEC
   lecLastFailureRespCode        = None                 Error code from last
                                                        failed Config/Join resp.
   lecLastFailureState           = Initial State        State of LEC when
                                                        updating above field.
   lecProtocol                   = 1                    Protocol specified by
                                                        LEC in Join requests.
   LecVersion                    = 1                    LEC Protocol Version
                                                        of above
   lecTopologyChange             = False
   lecConfigServerAtmAddress     = 00.00.00.00.00.00.

   lecConfigSource               = Did not use LECS
   lecActualLanType              = 802.3 - Ethernet     Frame format currently
                                                        used by LEC
   lecActualMaxDataFrameSize     = 1516
   lecActualLanName              = ETH                  Name of emulated LAN
                                                        that LEC joined.
```

```
lecActualLesAtmAddress        = 39.84.0F.00.00.00.

lecProxyClient                = False                 Is LES acting like a
                                                      proxy ?
```

super-elan

Example: super-elan

```
LEC+ mib super-elan
```

QoS Information

Use the **qos-information** command to get to the `LEC x QoS+` prompt from which you can monitor Quality of Service as described in “Quality of Service Console Commands” on page 51-1.

Syntax: qos-information

Example: `qos`

Exit

Use the **exit** command to return to the previous prompt level.

Syntax: exit

Example: `exit`

Appendix A. Quick Configuration Reference

Important

If you are attempting to configure or monitor your IBM 2210 and your service terminal is unreadable, see “Service Terminal Display Unreadable” in IBM 2210 Nways Multiprotocol Router Service and Maintenance Manual.

Quick Configuration Tips

Making Selections

On the panels that you view when using the Quick Configuration program, the information shown in brackets, [ ], is the default. For example:

```
Configure Bridging? (Yes, No, Quit): [Yes]
```

- To use the default `Yes`, press **Enter**.
- To use a value other than the default, such as `No` or `Quit`, choose from the values in the parentheses.
- If no value appears in the brackets, there is no default and you must type a value.

Exiting and Restarting

- To restart the current Quick Configuration section at any time, type **r**. For example, if you are in the Interface Configuration section, type **r** and press **Enter** to return to the beginning of that section.
- To exit Quick Configuration, type **q** and press **Enter**. The `Config>` prompt will appear.
- To restart Quick Configuration from the `Config>` prompt, type **qc** and press **Enter**.

When You’re Done

- Once you have completed your configuration, you must restart the IBM 2210 for the configuration to take effect. At the end of the Quick Configuration program, you are given this option.

Starting the Quick Configuration Program

The following sections describe sample configurations using the Quick Configuration program (**qconfig**).

To start the quick configuration program, enter **qc** at the `Config>` prompt.

The program displays the following panel after starting.

```
Router Quick Configuration for the following:

o   Interfaces
o   Multilink PPP (w/o DIALs)
o   Dial Circuits (w/o DIALs)
o   Dial-in Access to LANs (DIALs)
o   Bridging
        Spanning Tree Bridge (STB)
        Source Routing Bridge (SRB)
        Source Routing/Transparent Bridge (SR/TB)
        Source Routing Transparent Bridge (SRT)
o   Protocols
        IP (including OSPF, RIP, and SNMP)
        IPX
        DNA
o   Booting

Event Logging will be enabled for all configured subsystems
with logging level 'Standard'

Note: Please be warned that any existing configuration for a particular item
will be removed if that item is configured through Quick Configuration
```

Event logging records system activity, status changes, data transmission and reception, data and internal errors, and service requests. The logging level is set to standard (the default). For more information about error logging, refer to the *Event Logging System Messages Guide*.

During Quick Configuration you can:

1. Configure interfaces
2. Configure multilink PPP interfaces
3. Configure Dial circuits
4. Configure Dial-in and Dial-out circuits
5. Configure Dial-in Access to LANs (DIALs) information
6. Configure bridging
7. Configure protocols
8. Configure booting
9. Enable Console Modem-Control
10. Restart the router

Configuring LAN Emulation

If you added an ATM device, you will see the following prompts:

```
*********************************************************
LAN Emulation Configuration
*********************************************************

Type 'Yes' to Configure LAN Emulation
Type 'No' to skip LAN Emulation Configuration
Type 'Quit' to exit Quick Config

Configure LAN Emulation? (Yes, No, Quit): [Yes]
```

Configuring Interfaces

```
*********************************************
Interface Configuration
*********************************************

Type 'Yes' to Configure Interfaces
Type 'No' to skip Interface Configuration
Type 'Quit' to exit Quick Config

Configure Interfaces? (Yes, No, Quit): [Yes]
```

1. Take one of the following actions:
   - Enter **y** to display the interface configuration prompts.
   - Enter **n** to skip interface configuration and continue with quick configuration.
   - Enter **q** to exit quick configuration. This displays the `Config>` prompt. To restart quick configuration from this prompt, enter **qc**.

When interface configuration begins, you can type 'r' any time at this level to restart Interface Configuration

The only WAN interfaces that you can configure using Quick Config are PPP, Frame Relay, and V34. The only parameters you can configure for PPP and Frame Relay are the cable type and the line speed if the IBM 2210 is providing the clocking. For V34 interfaces the cable type is set to RS-232 DTE with a clock speed of 115200.

Note: Some modems do not support 115200 as the DTE serial speed. If this is the case, you must go into the network configuration for that V34 net and lower the DTE speed.

What quick configuration displays next depends on whether you have an Ethernet or Token-Ring version of the IBM 2210.

Ethernet

For Ethernet versions of the IBM 2210, configuration prompts similar to the following ones appear:

1. The interface verification:

```
Intf 0 is Ethernet

Intf 1 is WAN PPP
Encapsulation for WAN 1 (PPP, Frame Relay, V34):  [PPP] PPP
```

2. Enter one of the following values to specify the encapsulation type:

   ppp Point-to-Point Protocol
   fr Frame Relay
   V34 V.34 Modem Handler

   The following message is displayed for PPP and Frame Relay:

```
Cable type (RS-232 DTE, RS-232 DCE, V.35 DTE, V.35 DCE, V.36 DTE, X.21 DTE,
X.21 DCE: [RS-232 DTE] V.35 DCE
```

Note: DTE cable types are used when attaching to a modem or DSU. DCE cable types are used when connecting directly to another DTE device and you want the 2210 to provide the clocking.

3. Enter the cable type you have or will connect to this WAN port.

```
Internal clock speed (decimal) (2400 - 2048000): [0] 1544000
```

Internal Clock Speed appears only if you enter a DCE cable.

The WAN prompts repeat for WAN Port 2.

```
Intf 2 is WAN PPP
Encapsulation for WAN 2 (PPP, Frame Relay, V34): [PPP]
Cable type (RS-232 DTE, RS-232 DCE, V.35 DTE, V.35 DCE, V.36 DTE, X.21 DTE,
X.21 DCE: [RS-232 DTE] V.35 DCE
This is all configured device information:

Intf 0 is Ethernet, Connector (10BaseT, AUI) autoconfigured
Intf 1 is WAN 1 with PPP Encapsulation, V.35 direct attach cable
  internal clock speed 1544000 bits/second
Intf 2 is WAN 2 with PPP Encapsulation, V.35 modem cable

Save this configuration?  (Yes, No): [Yes]
```

4. Enter **y** to save the configuration and continue with quick configuration. Enter **n** to redisplay the interface configuration prompts.

Token-Ring

For token-ring versions of the IBM 2210, configuration prompts similar to the following ones appear.

1. The interface verification:

```
Intf 0 is Token Ring
Speed in Mb/sec (4,16): [16]
```

2. Enter **4** or **16** to specify the media transfer speed in megabits per second. The media transfer speed must match the speed of the ring.

```
Connector (STP, UTP): [STP]
```

3. Enter one of the following values to specify the media you are using:

   STP shielded twisted pair
   UTP unshielded twisted pair

   For a description of WAN prompts, see the Ethernet configuration prompts.

```
Intf 1 is a WAN PPP
Encapsulation for WAN 1
(PPP, Frame Relay, V34): [PPP]
Cable type (RS-232 DTE, RS-232 DCE, V.35 DTE, V.35 DCE, V.36 DTE, X.21 DTE,
X.21 DCE: [RS-232 DTE] V.35 DCE
Intf 2 is a WAN PPP
Encapsulation for WAN 2
(PPP, Frame Relay, V34): [PPP]
Cable type (RS-232 DTE, RS-232 DCE, V.35 DTE, V.35 DCE, V.36 DTE, X.21 DTE,
X.21 DCE: [RS-232 DTE] V.35 DCE
Internal clock speed (decimal) (4800 - 2048000): [0]

This is all configured device information:

Intf 0 is Token-Ring, Speed 16 Mb/sec, Connector UTP
Intf 1 is WAN1 with PPP Encapsulation, V.35 modem cable
Intf 2 is WAN2 with PPP Encapsulation, V.35 direct attach cable
   internal clock speed 0 bits/second

Save this configuration?  (Yes, No): [Yes]
```

4. Enter **y** to save the configuration and continue with quick configuration. Enter **n** to redisplay the interface configuration prompts.

Configuring Multilink PPP (MP) Interfaces

If you have a router with ISDN capabilities, the following configuration questions will be displayed.

Note: The following example assumes a Primary ISDN adapter plugged into a 2210 Model 24x or Model 14x.

```
*********************************************************
Multilink PPP Configuration (w/o DIALs)
*********************************************************

 Type 'Yes' to Configure Multilink PPP
 Type 'No' to skip Multilink PPP Configuration
 Type 'Quit' to exit Quick Config

 Configure Multilink PPP? (Yes, No, Quit): [Yes]
```

1. Take one of the following actions:
   - Enter y to display the Multilink PPP configuration prompts
   - Enter n to skip Multilink PPP configuration and continue quick configuration
   - Enter q to exit quick configuration

   The following status message appears when MP configuration begins displaying the current MP configuration. You have the choice of editing an existing MP interface configuration or starting a new MP bundle.

```
Current Multilink PPP Configuration:
Num    Intf#    Direction  Max Links  Link Intf#  Base Intf#  Destination
1      New Multilink PPP

Choose the Multilink PPP you wish to edit/add:  (1 - 1): [1]
```

2. Select the number of your choice. Enter the last number in the list to start a new MP interface configuration or select the number of an existing MP interface to modify the configuration. (Note: There are no existing MP interfaces in the

example above.) If you choose to add a new MP interface, the following questions will be asked. The questions vary slightly for INBOUND and OUTBOUND MP interfaces:

```
Enter maximum number of active links (2 - 23): [2] 3
Set Call Direction (Inbound, Outbound, Both): [Inbound] Inbound
Enter Idle timer (seconds, 0 means always active) (0 - 65535): [0] 0
```

3. Next you are prompted to add/edit the ISDN dial-circuits that can be used by the MP interface. The example below demonstrates adding one dial-circuit but you may add more than one dial-circuit per MP interface. Choose to add a dial-circuit by selecting the last number in the list denoted by "New Circuit" or to edit an existing dial-circuit configuration by typing its corresponding number. (Note: The example below does not display any existing dial-circuit configuration.)

```
Current Dial Circuit Configuration:

Num Intf# Intf Type               BaseIntf# MP  Direction  Destination
1   New Circuit

Choose a Dial Circuit Link you wish to edit/add:  (1 - 1): [1]
Enter interface # of Base Net, "?" for List,"Q" to quit: (6)

Address assigned name             Network Address  Network Subaddress
----------------------            ---------------  ------------------
default_address                   9999999
Assign Line ID *In* Network Address:
     Network Address name ([1-23] chars): LID_IN
     Enter Network Address [1-26 digits]: 1234
     Enter Network Subaddress [0-21 digits]:


Interface #:                8
Interface Type:             PPP Dial Circuit
Base Intferface #:          6 (ISDN Base Net)
Multilink PPP Interface #:  7
Call Direction:             Inbound only
Destination Name:           default_address
Line ID *IN* Name:          LID_IN

Is this correct (Yes, No): [Yes] Yes
Add another Dial Circuit Link (Yes, No): [Yes] No
```

4. Next, the MP interface and all of the dial-circuits for the interface are listed for confirmation. In this case, there is only one dial-circuit for the MP interface.

```
 Multilink PPP Interface #:  7
 Call Direction:             Inbound only
 Idle timer:                 0 (fixed circuit)
 Maximum Number of links:    3
 Dial Circuit Link
      Interface #:           8
      Interface Type:        PPP Dial Circuit
      BASE Interface #:      6 (ISDN Base Net)
      Destination Name:      default_address
      Line ID *In* Name:     LID_IN
Is this correct (Yes, No): [Yes] Y
```

5. In order to add/edit another MP interface type y to the following question. Answering n will exit you from the MP configuration section.

```
Add another Multilink PPP Interface (Yes, No): [Yes] n
```

6. After configuring all of the MP interfaces, an MP confirmation screen will appear with all of the configured MP interfaces listed. You can type y to save the changes or n to discard the new MP configuration.

```
Current Multilink PPP Configuration:
Num    Intf#    Direction  Max Links  Link Intf#  Base Intf#  Destination
1      7        In         3          8           6           default_ad

Save this configuration (Yes, No): [Yes] y

Multilink PPP configuration saved.
```

Configuring Dial-Circuits

The following configuration questions are displayed for dial-circuit configuration:

```
*********************************************************
Dial Circuit Configuration (w/o DIALs)
*********************************************************

Type 'Yes' to Configure Dial Circuits
Type 'No' to skip Dial Circuits Configuration
Type 'Quit' to exit Quick Config

Configure Dial Circuits? (Yes, No, Quit): [Yes] y
```

1. Take one of the following actions:
   - Enter y to display the Dial-Circuit configuration prompts
   - Enter n to skip Dial-Circuit configuration and continue quick configuration
   - Enter q to exit quick configuration

   The following status message appears upon entering the dial-circuit configuration. Note that in this example there is no existing dial- circuit configuration:

```
Current Dial Circuit Configuration:
Num Intf# Intf Type               BaseIntf# MP  Direction ...
Destination
1   New Circuit

Choose the circuit you wish to edit/add:  (1 - 1): [1] 1
```

2. Choose to add a new dial-circuit by selecting the number at the bottom of the list denoted by "New Circuit". Choose to edit an existing dial-circuit configuration by selecting the number of the dial-circuit which you wish to edit (Note: in the above example, there are no existing dial-circuits). The following is an example of the prompts that will be displayed to add a new, PPP, inbound dial-circuit:

```
Enter interface # of Base Net, "?" for List,"Q" to quit: (6)
Enter type of dial circuit for this net: (PPP, FRAME-RELAY): [FRAME-RELAY] PPP

Set Call Direction (Inbound, Outbound, Both): [Both] Inbound
Accept ANY INBOUND call (Yes, No): [No] Yes
```

3. After answering all of the questions, you will be given a confirmation for the dial-circuit as shown below:

```
Interface #:                13
Interface Type:             PPP Dial Circuit
Base Intterface #:          6 (ISDN Base Net)
Idle timer:                 0 (fixed circuit)
Call Direction:             Inbound only
Destination Name:           default_address
Line ID *IN* Name:          * ANY *

Is this correct (Yes, No): [Yes] Yes
```

4. Next, you may choose to add/edit more dial-circuits in the same way as the example above.

```
Add another Dial Circuit (Yes, No): [Yes] No
```

5. Finally, you will be asked to confirm the dial-circuit configuration and exit the dial-circuit configuration section. Answering y will save the dial-circuit configuration and answering n will discard changes made during this configuration session.

```
Current Dial Circuit Configuration:
Num Intf# Intf Type               BaseIntf# MP  Direction
Destination
1   13    PPP Dial Circuit        6/ISDN    No  In
default_addre

Save this configuration (Yes, No): [Yes] Yes

Dial circuit configuration saved.
```

Configuring Dial-in Access to LANs (DIALs) Interfaces and DIALs Server Information

If the router you are configuring contains the DIALs feature, then you will be asked if you want to configure DIALs interfaces and DIALs server information. You will only be asked to configure DIALs interfaces if you have configured V34 on a base WAN interface or if an ISDN interface exists in your router. The following prompts lead you through the DIALs configuration:

```
*********************************************************
Dial-in Access to LANs (DIALs) Configuration
*********************************************************

Type 'Yes' to Configure DIALs Configuration
Type 'No' to skip DIALs Configuration Configuration
Type 'Quit' to exit Quick Config

Configure DIALs Interfaces? (Yes, No, Quit): [Yes]
```

1. Take one of the following actions:
   - Enter y to display the DIALs Interface prompts
   - Enter n to skip DIALs Interface configuration
   - Enter q to exit quick configuration

   If you answer yes and there ISDN is loaded on this device, the following screen will be shown.

```
Current Multilink PPP Configuration:
Num  Intf#   Direction   MaxLinks  DIALs
1    8       In          2         No

Enter the number of Multilink PPP DIALs interfaces:(0-23) 2
Enter maximum number of active links per Multilink PPP interface: 3
```

Next, the following prompt will be shown.

```
For Base Interface #1 (V.34 Base Net) no Dial Circuits are configured!
Add a DIALs (Dial-in) Interface for this Base Interface? (Yes, No): [No]y
Add a Dial-out DIALs Interface for this Base Interface? (Yes, No): [No] y

Num Intf# Intf Type                BaseIntf#  MP  Direction  Destination

 1   3     PPP Dial-in Circuit      1/V34     No  In         N/A


 2   4     Dial-out Dials Circuit   1/V34     No  Out        N/A


Save this configuration (Yes, No): [Yes]

Dial circuit configuration saved.
```

Answering no will take the user out of the DIALs server configuration.

2. For every valid base WAN interface (V34 or ISDN) in the router, you are asked if you want to add a DIALs dial-in interface for this base net.
   - If the base net is ISDN BRI or ISDN PRI, you are asked if you want to add up to 2 or 23 respectively dial-in interfaces for the ISDN base net.
   - If the base net is V34 then you will also be asked if you want to add a DIALs dial-out circuit for this base net (Dial-out is not supported over ISDN).
3. After answering yes or no to these questions, the current dial-circuit configuration for that base net is displayed. You can then save the configuration by answering yes or restart the configuration for that base net by answering no.
4. After configuring all of your DIALs interfaces or by answering no to the DIALs interfaces question, you arrive at the DIALs Server configuration. Here you are asked to enter information about global settings for the DIALs server.

```
Configure DIALs Server? (Yes, No, Quit): [Yes] yes
Type 'r' any time at this level to restart Dial-in Access to LANs
Configuration.
```

5. Take one of the following actions:
   - Enter y to display the DIALs Server prompts
   - Enter n to skip DIALs Server configuration
   - Enter q to exit quick configuration

   If you answer yes, the following prompt will be shown. Answering no takes you to the next configuration section.

```
Default number of minutes a user is allowed before being
disconnected, 0 is unlimited: (0)
```

6. The default number of minutes on-line determines the maximum connection time for dial-in and dial-out users. Enter 0 if you want to this time to be unlimited. The default is zero if you have not configured this information previously.

```
Enter DIALs Server name - up to 30 chars: (2210_DIALS_SERVER)
```

7. Enter the name of the DIALs server. The default is 2210_DIALS_SERVER. This is the name of the server that will be displayed when dial-out clients "discover" DIALs Dial-out Servers on the network when they invoke the DIALs client's CHOOSER application.

```
Dial-out client type(s) supported (DIALs, TELNET, BOTH): [BOTH]
```

8. The previous question determines what level of dial-out support is turned on for the router. DIALs refers to supporting the IBM DIALs dial-out clients. Telnet dial-out refers to the ability to dial-out from a LAN based client using either a telnet application or a telnet serial port application. The default setting is to have both enabled.

```
Inactive time before a connection is dropped, 0 is unlimited: (30)
```

9. The previous question pertains to how long a dial-out circuit is active while no data is being transmitted or received. It should be set to the number of minutes that a connection over a dial-out circuit can be active without traffic. The default is 30 minutes.

```
Configure Proxy DHCP? (Yes, No, Quit): [Yes]
How many DHCP Servers do you wish to use? (Maximum is 20) : (1) 2
Enter DHCP Server Address: [] 10.0.0.1
Enter DHCP Server Address: [] 10.0.0.2
```

10. The DHCP Gateway interface, or giaddr (as defined in RFC1531), is the IP address associated with the subnet you wish the DHCP server to offer addresses. This is necessary because the DHCP server may be used to lease addresses to more than one subnet. The giaddr allows the DHCP server to distinguish which subnet to offer addresses, as well as provide an address in which to respond to.

    Quick Config will now ask you for the number of the interface that you plan to configure as the subnet associated with your dial-in users. If you have only one LAN interface, the number of that interface is most likely zero.

```
 DHCP Gateway (giaddr) interface: (0)

Do you want to use Dynamic DNS with your DHCP server? (Yes, No): [Yes]
```

11. The next set of questions determine your Dynamic Host Configuration Protocol (DHCP) configuration.

    If you will be using DHCP to administer IP address to your dial-in users, you should answer yes to this question. If you answer yes, then you will be asked to enter the DHCP server addresses and the network number that is connected to the LAN your dial-in users are trying to access.

```
This is all the configured Dial-in Access to LANs information:

Default number of minutes allowed per connection:  15
Inactive timer:                                    Unlimited
LAN Protocols enabled for dial-out:                TELNET SHIVA
DIALs Server name:                                 2210_DIALS_SERVER

DIALs client IP address specification:
Client     : Disabled
UserID     : Disabled
Interface  : Disabled
DHCP Proxy : Enabled

Configured DHCP Servers :           10.0.0.1          10.0.0.2

DHCP Gateway (giaddr) interface: 0
Lease addresses will be associated with the
network (subnet) accessed via 10.0.0.2

Dynamic DNS: Enabled


Is this information correct? (Yes, No, Quit): [Yes]
```

12. A summary of the information for DIALs configuration is displayed and you are asked if it is correct. If the information is correct, answer yes. If it is not and you want to re-enter the information, answer no. If you want to terminate quick config, answer quit.

Configuring Bridging

```
*********************************************
Bridging Configuration
*********************************************

Type 'Yes' to Configure Bridging
Type 'No' to skip Bridging Configuration
Type 'Quit' to exit Quick Config

Configure Bridging? (Yes, No, Quit): [Yes]
```

1. In response to `Configure Bridging`, take one of the following actions:
   - Enter **y** to display the bridging configuration prompts. The prompts that appear depend on your network configuration.
   - Enter **n** to skip the bridging configuration and continue with quick configuration.
   - Enter **q** to exit quick configuration. This displays the `Config>` prompt. To reenter quick configuration, enter **qc** after this prompt.
2. If you have configured for DIALs dial-in circuits the following panel will be displayed:

```
Transparent bridging automatically enabled
on DIALs ports? (Yes, No, Quit): [Yes]
```

   Enter **y** to automatically add transparent bridge ports to the bridge configuration for each of the DIALs interfaces.

   Enter **n** to automatically disable Bridging on each of the DIALs dial-in interfaces.

3. If you choose to configure bridging, Spanning Tree Bridging (STB) will be enabled on all LAN interfaces. You will see the following panels:

```
Type 'r' any time at this level to restart Bridging Configuration

STB will be enabled on all LAN interfaces
```

   Enter **y** to configure SRT bridging. Otherwise, enter **n**. For each Token-Ring interface in the configuration, you will be prompted to enable Source Routing on the interface.

```
Configure SRT Bridging? (Yes, No): [Yes]
You are now configuring the Source Routing part of SRT Bridging
Bridge Number (hex) of this Router (1-F): [A]
```

4. Enter a bridge number, which is a hexadecimal value from 1 to F that is unique between two parallel segments.

```
Interface 0 (Port 1) is of type Token Ring
Configure Source Routing on this interface (Yes, No): [Yes]
```

5. Enter **y** to configure source routing on the interface. The console displays the next two lines.

```
Configuring Interface 0 (Port 1)
Segment Number (hex) of this Interface (1-FFF): [A1]
```

   Note: The port number increases by one because source routing bridging does not allow a port number of zero.

   A unique hexadecimal value from 1 to FFF is assigned to each interface. The interfaces on each ring (segment) have the same segment number, but the segment number is unique to each ring.

   These prompts appear for each Token Ring interface.

```
Interface 1 (Port 2) is of type Token Ring
Configure Source Routing on this interface? (Yes, No): [Yes]
Configuring Interface 1 (Port 2)
Segment Number (hex) of this Interface (1-FFF): [A2]
```

   If more than two interfaces are configured for source routing, enter a unique hexadecimal value from 1 to FFF unique for the internal virtual segment.

```
Virtual Segment Number (hex) of this Router (1-FFF): [A4]
```

6. A panel similar to the following is displayed:

```
This is all configured bridging information:

       Interfaces configured for STB:

       Interface #    Port #     Interface Type

           0          1          Token Ring
           1          2          Token Ring

The Source Routing part of SRT Bridging has been enabled

Bridge Number of this Router: A

Interfaces configured for Source Routing:

       Interface #     Port#      Segment #   Interface Type

           0             1          A1        Token Ring
           1             2          A2        Token Ring

Virtual Segment Number of this Router:  A4

Save this Configuration? (Yes, No): [Yes]
```

7. Enter **y** to save the bridging configuration and continue with quick configuration. Enter **n** to redisplay the bridging configuration prompts.

   If you enter **y**, the following message appears:

```
Bridging configuration saved
```

Configuring Protocols

After you save the bridging configuration, you will see the following panel:

```
*********************************************
Protocol Configuration
*********************************************

Type 'Yes' to Configure Protocols
Type 'No' to skip Protocol Configuration
Type 'Quit' to exit Quick Config

Configure Protocols? (Yes, No, Quit): [Yes]
```

Take one of the following actions:

- Enter **y** to configure the protocols.
- Enter **n** to skip protocol configuration and continue with quick configuration.
- Enter **q** to exit quick configuration.

You will first configure IP, then IPX, and then DECnet.

Configuring IP

When you answer **y** to the Configure Protocol panel, quick configuration displays the following messages:

```
Type 'r' any time at this level to restart Protocol configuration

Configure IP? (Yes, No): [Yes]
```

1. Take one of the following actions:
   - Enter **y** to configure IP.
   - Enter **n** to skip IP configuration and continue with quick configuration.

If you have configured for DIALs dial-in interfaces, the following panel will be displayed:

```
Automatically configure IP on DIALs dial-in interfaces (this will
also enable ARP subnet routing)? (Yes, No, Quit): [Yes]
```

2. Take one of the following actions:
   - Enter **y** to automatically add un-numbered IP addresses for each DIALs interface. It will also enable ARP Subnet Routing for the router and turn off the sending of RIP packets on DIALs interfaces. All of these options are required for Dial-In Access to LANs interfaces and it is recommended for you to answer yes to this question if you desire IP to be enabled on DIALs interfaces.
   - Enter **n** to automatically disable IP on each of the DIALs dial-in interfaces.

The following lines appear for each interface.

```
Configuring Per-Interface IP Information

Configuring Interface 0 (Token Ring)
Configure IP on this interface? (Yes, No): [Yes]
IP Address: [ ] 128.185.141.1
Address Mask: [255.255.0.0]
```

3. Enter the IP address in decimal notation for example, 128.185.142.20. The console displays one of the following error messages if you enter an invalid IP address:

   ```
   Bad address, please try again.

   This address has already been assigned.  Enter a different address
   ```

   Address mask is a decimal value that reflects the IP network or subnetwork to which this interface is attached.

   For more information about IP addressing or address masks, refer to the *Protocol Configuration and Monitoring Reference*, or consult your network administrator.

```
Per-Interface IP Configuration complete

Configuring IP Routing Information
Enable Dynamic Routing (Yes, No): [Yes]
```

4. Enter **y** if you want the routing protocols (RIP or OSPF) to build the routing tables. Enter **n** to manually add IP address destinations to the routing tables (static routes).

```
Enable OSPF? (Yes, No): [Yes]
```

5. Enter **y** to enable the OSPF routing protocol as the primary dynamic IP routing protocol. RIP will be enabled only to send advertisements, not to receive them. Enter **n** if you do not want to use OSPF. RIP will be enabled to send and receive advertisements.

```
OSPF Enabled with Max routes = 1000  and Max routers = 50
```

   Max routes is the maximum number of autonomous system (AS) external routes imported into the OSPF routing domain. Max routers is the maximum number of OSPF routers in the routing domain.

```
Routing Configuration Complete

SNMP will be configured with the following parameters:

Community: public
Access:    READONLY

If you plan to use the graphical configuration tool
to download a configuration, it requires the definition
of a community name with read_write_trap access.

Define community with read_write_trap access ? (Yes, No):  [Yes]


This is the information you have entered:

          Interface #        IP Address     Address Mask

             0               128.185.141.1  255.255.255.0
             1               128.185.142.1  255.255.255.0
             2               128.185.143.1  255.255.255.0

OSPF is configured, and RIP is configured only for 'sending'

SNMP has been configured with the following parameters:

     Community: public
     Access:    read_trap

     Community: dana
     Access:    read_write_trap

Save this configuration? (Yes, No): [Yes]
```

6. Enter **y** to save the IP configuration and continue with quick configuration. Enter **n** to redisplay the protocol configuration prompts.

Configuring IPX

After you save the IP configuration, you will see the following messages:

```
Configure IPX? (Yes, No): [Yes]
```

1. Enter **y** to configure IPX. Enter **n** to skip IPX configuration and continue with quick configuration.

You will see messages similar to the following:

```
Type 'r' any time at this level to restart IPX Configuration
IPX Configuration is already present
Configure IPX anyway? (Yes, No): [No] yes
```

2. Enter **y** to replace the existing configuration. Enter **n** to keep the current configuration and continue.

If you have configured for DIALs dial-in interfaces the following panel will be displayed:

```
Enable IPX on DIALs interfaces? (Yes, No): [Yes]
```

3. Enter **y** to automatically enable IPX on each of the DIALs interfaces. A random IPX network number will be generated for the interface and IPXWAN will be disabled for the DIALs interface. It is required that IPXWAN be disabled for DIALs interfaces.

   Enter **n** to automatically disable IPX on each of the DIALs dial-in interfaces.

```
Configuring Per-Interface IPX Information

Configuring Interface 0 (Token Ring)
Configure IPX on this interface? (Yes, No): [Yes]
```

4. The next messages and your responses depend on whether you are configuring Token-Ring or Ethernet.

   Configuring IPX for Token-Ring:

   a. The following prompt is displayed:

```
Token Ring encapsulation (frame) type? (TOKEN–RING MSB, TOKEN–RING LSB,
    TOKEN–RING_SNAP MSB, TOKEN–RING_SNAP LSB): [TOKEN–RING MSB]
```

   b. Enter the encapsulation type used by the IPX protocol on your Token-Ring end stations.

`Token–Ring MSB:`	Most common encapsulation type and the default. The IBM 2210 builds outgoing packets with a 3-byte 802.2 header, (0xE0, 0xE0, 0x03). It sends the source and destination addresses in MSB (most significant bit), or noncanonical, format, which is the native address format for Token-Ring.
`Token–Ring LSB`	Same as Token-Ring MSB except the IBM 2210 sends the addresses in LSB (least significant bit), or canonical, format.
`Token-Ring SNAP MSB`	The IBM 2210 builds outgoing packets with an 8-byte 802.2/SNAP header (0xAA, 0xAA, 0x03, 0x00, 0x00, 0x00, 0x81, 0x37). It sends the source and destination addresses in most significant bit (MSB), or noncanonical, format.
`Token-Ring SNAP LSB`	Same as Token-Ring SNAP MSB except the IBM 2210 sends the addresses in LSB, or canonical, format.

Configuring IPX for Ethernet:

a. The following prompts are displayed:

```
Ethernet encapsulation type? (ETHERNET_8022, ETHERNET_8023, ETHERNET_ii,
ETHERNET_SNAP): [ETHERNET_8023]
```

b. Enter the encapsulation type used by the IPX protocol on your Ethernet end stations.

Ethernet_8022	Packet includes an 802.2 header.
Ethernet_8023	Uses an IEEE 802.3 packet format without the 802.2 header. This is the default and the default for NetWare versions prior to 4.0. Ethernet 802.3 does not conform to the IEEE 802 standards because it does not include an 802.2 header. It may cause problems with other nodes on the network.
Ethernet_II	Uses Ethernet type 8137 as the packet format. This format is required if you are using NetWare VMS on the Ethernet. This is the default for Netware Versions 4.0 and higher.
Ethernet_SNAP	Uses the 802.2 format with a SNAP header. This encapsulation type is meant to be compatible with token-ring SNAP encapsulation. However, it violates IEEE standards and is not interoperable across conformal bridges.

```
Network Number (hex) (1-FFFFFFFD):[1] 1
```

5. Assign an IPX network number to the associated directly connected network. Every IPX interface must have a unique network number.

```
Configuring Interface 1 (WAN PPP)
Configure IPX on this interface? (Yes, No): [Yes]
Network Number (hex) (1-FFFFFFFD): [1] 2

Enable IPXWAN? (Yes, No): [No] yes

Configuring Interface 2 (WAN PPP)
Configure IPX on this interface? (Yes, No): [Yes]
Network Number (hex) (1-FFFFFFFD):[1] 3

Enable IPXWAN? (Yes, No): [No] yes

Host Number for Serial Lines: (000000000000) 1

Configure IPXWAN NodeID? (Yes, No): [Yes]
NodeID (hex) (1 - FFFFFFFD): [1] 4
```

   If enabled, the IPXWAN protocol negotiates routing parameters to be used on the PPP serial interface before IPX packet forwarding begins. IPXWAN is not required to forward IPX packets on PPP serial interfaces. The IPXWAN Node ID is a unique IPX network number that identifies the router, and is required if IPXWAN is enabled on any network interfaces.

6. Host number is a unique 12-digit hexadecimal value assigned to an IPX router. It is required because serial lines do not have hardware node addresses from which to build a host number.

```
This is the information you have entered:

               Per-Interface Configuration Information

Ifc   IPX Net (hex)   Encapsulation       IPXWAN

0     1               TOKEN-RING MSB      Not Configured
1     2                                   Enabled
2     3                                   Enabled

  Host Number for Serial Lines: 000000000001
  IPXWAN Node ID = 4
  IPX Router Name = ipx_router-4
Save this configuration? (Yes, No): [Yes]
```

7. Enter **y** to save the IPX configuration and continue with quick configuration. Enter **n** to redisplay the IPX configuration prompts.

   If you enter **y**, the following message appears:

```
IPX configuration saved
```

Configuring DECnet (DNA)

After you save the IPX configuration, you will see the following messages.

```
IPX Configuration saved

Configure DNA? (Yes, No): [Yes]
```

1. Enter **y** to configure DNA. Enter **n** to skip DNA configuration and continue with quick configuration.

```
Type 'r' any time at this level to restart DNA Configuration

Configuring Global DNA information

Highest Node Number (decimal) (1-1023): [32]
Router Level (Level1, Level2, DEC Level1, DEC Level2):
  [ Level2]
Highest Area (decimal) (1-63): [63]
Node Address (area.node): (63.32)
```

   The above configuration fields are configured with the following considerations:

   Highest Node Number Is the highest node address in the router's area. Setting it excessively high will affect the routers efficiency and require excess storage.

   Router Level Identifies whether the router is a Level 1 or Level 2 router. A Level 1 router keeps track of all nodes in its area and does not care about nodes outside its area. A Level 2 router routes traffic between areas.

   Normally you should select Level1 or Level2 with the following exception: select DEC Level1 or DEC Level2 only when this router must communicate over X.25 networks with routers conforming to the DEC X.25 standard.

Highest Area This number should be at least as high as the highest area number in the overall network.

Node Address Is the node ID of this router and must be unique in the network.

When you press Enter, the following is displayed:

```
Configuring Per-Interface DNA Information

Configuring Max Routers on each interface

Configuring Interface 0 (Ethernet)
Configure DNA on this interface? (Yes, No) [YES]
Max Routers (decimal) (1-33): [16]

Configuring Interface 1 (WAN PPP)
Configure DNA on this interface? (Yes, No) [Yes]

Configuring Interface 2 (Token Ring)
Configure DNA on this interface? (Yes, No) [Yes]
Max Routers (decimal) (1-33): [16]
```

2. Enter **y** for every interface that will be connected to the DECnet network. For LANs, Max Routers specifies how many other routers may be on this circuit. For router efficiency and memory requirements set this argument to a few more than the total number of adjacent routers on this circuit.

   The following panel is displayed:

```
This is the information you have entered:

        Global Configuration Information

           Highest Node Number:       32
           Router Level:              Level2
           Highest Area:              63
           Node Address:              63.32

        Pre-Interface Configuration Information
      Interface Number           Max Routers

              0                          16
              1                          1
              2                          16

Save this configuration? (Yes, No): [Yes]
```

3. Enter **y** to save the DECnet configuration and continue with the quick configuration. Enter **n** to redisplay the DECnet configuration prompts.

   If you enter **y**, the following message appears:

```
DNA Configuration Saved
```

Configuring Booting

```
*******************************************
Boot Configuration
*******************************************

Type 'Yes' to Configure Booting
Type 'No' to skip Booting Configuration
Type 'Quit' to exit Quick Config

Configure Booting? (Yes, No, Quit): [Yes]
```

1. Enter **y** to display the boot configuration prompts. Enter **n** to skip boot configuration. Enter **q** to exit quick configuration.

   Any previous boot information is displayed, as illustrated in the following example:

```
Type 'r' any time at this level to restart Boot configuration


Previous Boot information

Booting Method:TFTP Boot
Interface Number:0
Interface IP Address:128.185.133.18
Address Mask:255.255.255.0
Host IP Address:128.185.120.120
Gateway IP Address:128.185.133.7
Boot file Name:ibm2210.ldc
Create a boot record using this information? (yes, No): [Yes]
```

2. Enter **y** to create a boot record with the previous boot information and display the following prompts:

```
Boot Configuration saved

Enable Console Modem-Control  (Yes, No, Quit): [No]
```

3. Take one of the following actions:

   - Enter **y** if you are connecting a console to the IBM 2210 through a modem and if you want autologout on lost phone connections.
   - Enter **n** to connect a console directly to the IBM 2210.
   - Enter **q** to exit quick configuration.

   When you enter **no**, you can then select another boot option from the next prompt.

```
Select Booting Method (TFTP Boot, BOOTP Boot, IBD Boot): [ ]
```

4. Enter the booting method you will use to boot the IBM 2210:

   - TFTP
   - BOOTP
   - IBD

The following sections describe the prompts that appear for each method.

TFTP Boot

```
Select Booting Method (TFTP Boot, BOOTP Boot, IBD Boot): [ ]
```

1. Enter **TFTP** to boot using a TFTP host server and respond to the following prompts:

`Interface Number ( ):[0]`	The number of the LAN interface over which to boot. For this version of the IBM 2210, you must use the default of 0.
`Interface IP Address: [0.0.0.0]`	IP address of the interface over which to boot. Enter the IP address in decimal notation.
`Address Mask: [255.255.0.0]`	Address mask identifies the IP address class type. Class A is 255.0.0.0, Class B is 255.255.0.0, and Class C is 255.255.255.0.
`Host IP Address:[ ]`	IP address of the host that contains the boot file.
`Via Gateway:[ ]`	If the host is not on the same (sub)network as the IBM 2210, enter the IP address of an intermediate router.
`Boot File Name:` (*/path/filename.ext*)	Name of the file over which to boot. You must use the full path for the boot file, for example: `/usr/2210/bootfile.name`

```
TFTP Boot Configuration Complete

This is the information you have entered:

       Booting Method:TFTP Boot
       Interface Number:0
       Interface IP Address:128.185.141.1
       Address Mask:255.255.255.0
       Host IP Address:128.185.120.120
       Gateway IP Address:128.185.141.7
       Boot File Name:ibm2210.ldc

Save this configuration? (Yes, No): [Yes]
```

2. Enter **y** to create a boot record. Enter **n** to restart the boot configuration prompts.

BOOTP Boot

```
Select Booting Method (TFTP Boot, BOOTP Boot, IBD Boot): [ ]
```

1. Enter **BOOTP** and the console displays a prompt to enter the interface number over which to boot.

   Then a message similar to the following appears:

```
BOOTP Boot Configuration Complete

This is the information you have entered:

       Booting Method:BOOTP Boot
       Interface Number:    1

Save this configuration? (Yes, No): [Yes]
```

2. Enter **y** to create a boot record. Enter **n** to restart the boot configuration prompts.

IBD Boot

```
Select Booting Method (TFTP Boot, BOOTP Boot, IBD Boot): [ ]
```

1. Enter **IBD** and the console displays a list of software loads in the IBD.

```
The following # loads(s) exist in the IBD
load.name load.name load.name load.name

You may use only these loads to configure an IBD boot record
IBD Load Name: (load.name) [ ]
```

2. Enter the name of the load you want the IBM 2210 to load when it boots.

```
IBD Boot Configuration Complete

This is the information you have entered:

       Booting Method:        IBD Boot
       IBD Load Name:         load.name
```

If a load does not exist in the IBD, you receive the following message:

```
There are no loads in the IBD. Select another booting
method
```

3. Enter **TFTP** or **BOOTP** to use another booting method.

Enabling Console Modem-Control

```
Enable Console Modem-Control  (Yes, No, Quit): [No]
```

Take one of the following actions:

- Enter **y** if you are connecting a console to the IBM 2210 through a modem and if you want autologout on lost phone connections.
- Enter **n** to connect a console directly to the IBM 2210.
- Enter **q** to exit quick configuration.

Restarting the Router

After configuring, you will receive the following message:

```
Quick Config Done
Restart the router? (Yes, No): [Yes]
```

1. Enter **y** to restart the router with the new configuration and display the following information:

```
RESTARTING THE ROUTER........

Copyright IBM Corp. 1994, 1996
MOS Operator Control

*
```

2. Enter **n** and the console displays the following message:

```
Type RESTART at the Config> prompt for the configuration to take effect
Config>
```

3. Enter **restart** after the `Config>` prompt to restart the IBM 2210 with the new configuration. To change or view the current configuration, enter **qc**.

Appendix B. X.25 National Personalities

This appendix lists the default settings for GTE-Telenet and DDN.

GTE-Telenet

The following parameters are the default settings for GTE-Telenet:

- Callreq: 20
- Clearreq:
  - Retries: 1
  - Timer: 18
- Disconnect: Passive
- DP-timer: 500 milliseconds
- Frame window size: 7
- Network Type: CCITT
- N2 timeouts: 20
- Packet:
  - Default size: 128
  - Maximum size: 256
  - Window size: 2
- Reset
  - Retries: 1
  - Timer: 18
- Restart
  - Retries: 1
  - Timer: 18
- Standard: 1984
- T1-timer: 4
- T2-timer: 2

DDN

The following parameters are the default settings for DDN:

- Callreq: 20
- Clearreq:
  - Retries: 1
  - Timer: 18
- Disconnect: Passive
- DP-timer: 500 milliseconds
- Frame window size: 7
- Network Type: CCITT

- N2 timeouts: 20
- Packet:
  - Default size: 128
  - Maximum size: 256
  - Window size: 2
- Reset
  - Retries: 1
  - Timer: 18
- Restart
  - Retries: 1
  - Timer: 18
- Standard: 1984
- T1-timer: 4
- T2-timer: 2

Appendix C. Making a Router Load File from Multiple Disks

If a software load arrives on multiple disks, use the procedure in the following sections to combine the loads into one load file that the router can use at the time of booting.

The first disk contains the following four files that you need if you want to fragment an existing load for transport on multiple diskettes.

cutup.c (UNIX C source file that can be compiled using a standard C compiler)

cutup.exe (DOS)

Use the following files for reassembling the load fragments onto a DOS or UNIX server.

kopy.bat (DOS)
kopy (UNIX shell script)

Assembling a Load File Under DOS

To assemble a load from the two diskettes, use the DOS batch file provided on diskette 1 (KOPY.BAT) using the following syntax:

kopy <*installation_drive***><***destination_directory***>**

Before assembling the load make sure that you have created a destination directory, and that you have inserted the first diskette in the drive specified by the installation_diskette_drive parameter. The following example illustrates this procedure.

```
B:\>kopy b: c:\source\cutup\tmp
B:\>copy c:\gw0/B c:\source\cutup\tmp\gw.tmp
1 file(s) copied
.
Please mount the second diskette
Press any key to continue . . .
Copying the second load file fragment
B:\>
B:\>copy c:\source\cutup\tmp\gw.tmp/B + b:\gw1
c:\source\cutup\tmp\gw.tmp  c:\SOURCE\CUTUP\TMP\GW.TMP
B:\GW1
1 file(s) copied
B:\>rename c:\source\cutup\tmp\gw.tmp  gw.ldc
Load file reassembly was successful
B:>
```

Assembling a Load File Under UNIX

To assemble a load from two UNIX diskettes, you can use the UNIX Bourne shell script (kopy) provided on diskette 1 using the following syntax:

kopy<*installation_drive*><*diskette_directory*><*destination_directory*

Before assembling the load make sure that you have created the mount and destination directories, and that you have inserted the first diskette in the drive specified by the installation_diskette_drive parameter. The following example illustrates this procedure.

```
kopy /dev/fd0 /kew /pcfs

Please insert the first diskette

Copying the first load file fragment

Please mount the second diskette

Copying the second load file fragment

Load file reassembly was successful

# ls /kew

gw0   gw1   gw.ldc
```

If you can't use the UNIX Bourne shell script, you can assemble the load manually using the following procedure:

1. Copy the load fragments on the two diskettes (gw0 and gw1) into a directory on the UNIX file system.
2. Type the following UNIX command:

   cat gw0 gw1 > gw.ldc

   The resulting file (gw.ldc) is the assembled router load.

Disassembling a Load File Under DOS

To disassemble a load under DOS, use the CUTUP.EXE file as follows:

cutup<*file_extension*><*file_name*><*cut_length*>

The file_extension is attached to the front of each slice needed to cut. The file_name is the DOS file name of the file to be disassembled. The cut_length is the length that CUTUP.EXE makes each fragment as it disassembles the file. The following example illustrates this procedure.

```
C: \source\cutup>dir
Volume in drive C has no label
Volume Serial Number is XXXXXXXX
Directory of C: \SOURCE\CUTUP
.0730934:46p
..0730934:46p
GW      LDC 10225660728931:22p
CUTUP   EXE   105410902939:38a
2 file(s)   1033107 bytes
14811136 bytes free
C: \source\cutup>cutup gw.ldc gw 1000000
.....................................................
.................
.........................
c: \SOURCE\CUTUP>dir
Volume in drive C has no label
Volume Serial Number is XXXXXXXX
Directory of C: \SOURCE\CUTUP
.0730934:46p
..0730934:46p
GW      0 10000000801931:22p
GW      LDC     10225660728931:22p
CUTUP   EXE   105410902939:38a
GW      1   225660801931:22p
4 file(s)   2055673 bytes
14811136 bytes free
```

Disassembling a Load File Under UNIX

To disassemble a load under use cutup.c. Begin by compiling the program using your UNIX compiler to make a cutup executable file. Then use the following syntax:

cutup<*file_extension*><*file_name*><*cut_length*>

The file_extension is attached to the front of each slice needed to cut. The file_name is the DOS file name of the file to be disassembled. The cut_length is the length CUTUP.EXE that is used to disassemble the file. The following example illustrates this procedure.

```
#  ls -la
total 658
drwxrxrx   2 root    512 Aug  114:41  .
drwxrxrx  26 root   1024 Aug  114:41  ..
drwxrxrx   2 root  24576 Aug  114:41  cutup
drwxrxrr   2 root1022566 Aug  114:41  gw.ldc

#  cutup gw.ldc gw 100000

#  ls -la

total 658
drwxrxrx   2 root    512 Aug  114:41  .
drwxrxrx  26 root   1024 Aug  114:41  ..
drwxrxrx   2 root  24576 Aug  114:41  cutup
drwxrxrr   2 root1022566 Aug  114:41  gw.ldc
drwxrxrr   2 root1000000 Aug  114:41  gw0
drwxrxrr   2 root  22566 Aug  114:41  gw1
```

List of Abbreviations

AARP AppleTalk Address Resolution Protocol

ABR area border router

ack acknowledgment

AIX Advanced Interactive Executive

AMA arbitrary MAC addressing

AMP active monitor present

ANSI American National Standards Institute

AP2 AppleTalk Phase 2

APPN Advanced Peer-to-Peer Networking

ARE all-routes explorer

ARI ATM real interface

ARI/FCI address recognized indicator/frame copied indicator

ARP Address Resolution Protocol

AS autonomous system

ASBR autonomous system boundary router

ASCII American National Standard Code for Information Interchange

ASN.1 abstract syntax notation 1

ASRT adaptive source routing transparent

ASYNC asynchronous

ATCP AppleTalk Control Protocol

ATP AppleTalk Transaction Protocol

AUI attachment unit interface

AVI ATM virtual interface

ayt are you there

BAN Boundary Access Node

BBCM Bridging Broadcast Manager

BECN backward explicit congestion notification

BGP Border Gateway Protocol

BNC bayonet Niell-Concelman

BNCP Bridging Network Control Protocol

BOOTP BOOT protocol

BPDU bridge protocol data unit

bps bits per second

BR bridging/routing

BRS bandwidth reservation

BSD Berkeley software distribution

BTP BOOTP relay agent

BTU basic transmission unit

CAM content-addressable memory

CCITT Consultative Committee on International Telegraph and Telephone

CD collision detection

CGWCON Gateway Console

CIDR Classless Inter-Domain Routing

CIP Classical IP

CIR committed information rate

CLNP Connectionless-Mode Network Protocol

CPU central processing unit

CRC cyclic redundancy check

CRS configuration report server

CTS clear to send

CUD call user data

DAF destination address filtering

DB database

DBsum database summary

DCD data channel received line signal detector

DCE data circuit-terminating equipment

DCS Directly connected server

DDLC dual data-link controller

DDN Defense Data Network

DDP Datagram Delivery Protocol

DDT Dynamic Debugging Tool

DHCP Dynamic Host Configuration Protocol

dir directly connected

DL data link

DLC data link control

DLCI data link connection identifier

DLS data link switching

DLSw data link switching

DMA direct memory access

DNA Digital Network Architecture

DNCP DECnet Protocol Control Protocol

DNIC Data Network Identifier Code

DoD Department of Defense

DOS Disk Operating System

DR designated router

DRAM Dynamic Random Access Memory

DSAP destination service access point

DSE data switching equipment

DSE data switching exchange

DSR data set ready

DSU data service unit

DTE data terminal equipment

DTR data terminal ready

Dtype destination type

DVMRP Distance Vector Multicast Routing Protocol

E1 2.048 Mbps transmission rate

EDEL end delimiter

EDI error detected indicator

EGP Exterior Gateway Protocol

EIA Electronics Industries Association

ELAN Emulated LAN

ELAP EtherTalk Link Access Protocol

ELS Event Logging System

ESI End system identifier

EST Eastern Standard Time

Eth Ethernet

fa-ga functional address-group address

FCS frame check sequence

FECN forward explicit congestion notification

FIFO first in, first out

FLT filter library

FR Frame Relay

FRL Frame Relay

FTP File Transfer Protocol

GMT Greenwich Mean Time

GOSIP Government Open Systems Interconnection Profile

GTE General Telephone Company

GWCON Gateway Console

HDLC high-level data link control

HEX hexadecimal

HPR high-performance routing

HST TCP/IP host services

HTF host table format

IBD Integrated Boot Device

ICMP Internet Control Message Protocol

ICP Internet Control Protocol

ID identification

IDP Initial Domain Part

IDP Internet Datagram Protocol

IEEE Institute of Electrical and Electronics Engineers

Ifc# interface number

IGP interior gateway protocol

InARP Inverse Address Resolution Protocol

IP Internet Protocol

IPCP IP Control Protocol

IPPN IP Protocol Network

IPX Internetwork Packet Exchange

IPXCP IPX Control Protocol

ISDN integrated services digital network

ISO International Organization for Standardization

Kbps kilobits per second

LAN local area network

LAPB link access protocol-balanced

LAT local area transport

LCP Link Control Protocol

LED light-emitting diode

LF largest frame; line feed

LIS Logical IP subnet

LLC logical link control

LLC2 logical link control 2

LMI local management interface

LRM LAN reporting mechanism

LS link state

LSA link state advertisement

LSA

LSB least significant bit

LSI LAN shortcuts interface

LSreq link state request

LSrxl link state retransmission list

LU logical unit

MAC medium access control

Mb megabit

MB megabyte

Mbps megabits per second

MBps megabytes per second

MC multicast

MCF MAC filtering

MIB Management Information Base

MIB II Management Information Base II

MILNET military network

MOS Micro Operating System

MOSDDT Micro Operating System Dynamic Debugging Tool

MOSPF Open Shortest Path First with multicast extensions

MSB most significant bit

MSDU MAC service data unit

MTU maximum transmission unit

nak not acknowledged

NBMA Non-Broadcast Multiple Access

NBP Name Binding Protocol

NBR neighbor

NCP Network Control Protocol

NCP Network Core Protocol

NetBIOS Network Basic Input/Output System

NHRP Next Hop Resolution Protocol

NIST National Institute of Standards and Technology

NPDU Network Protocol Data Unit

NRZ non-return-to-zero

NRZI non-return-to-zero inverted

NSAP Network Service Access Point

NSF National Science Foundation

NSFNET National Science Foundation NETwork

NVCNFG nonvolatile configuration

OPCON Operator Console

OSI open systems interconnection

OSICP OSI Control Protocol

OSPF Open Shortest Path First

OUI organization unique identifier

PC personal computer

PCR peak cell rate

PDN public data network

PING Packet internet groper

PDU protocol data unit

PID process identification

P-P Point-to-Point

PPP Point-to-Point Protocol

PROM programmable read-only memory

PU physical unit

PVC permanent virtual circuit

RAM random access memory

RD route descriptor

REM ring error monitor

REV receive

RFC Request for Comments

RI ring indicator; routing information

RIF routing information field

RII routing information indicator

RIP Routing Information Protocol

RISC reduced instruction-set computer

RNR receive not ready

ROM read-only memory

ROpcon Remote Operator Console

RPS ring parameter server

RTMP Routing Table Maintenance Protocol

RTP RouTing update Protocol

RTS request to send

Rtype route type

rxmits retransmissions

rxmt retransmit

SAF source address filtering

SAP service access point

SAP Service Advertising Protocol

SCR Sustained cell rate

SCSP Server Cache Synchronization Protocol

sdel start delimiter

SDLC SDLC relay, synchronous data link control

seqno sequence number

SGID sever group id

SGMP Simple Gateway Monitoring Protocol

SL serial line

SMP standby monitor present

SMTP Simple Mail Transfer Protocol

SNA Systems Network Architecture

SNAP Subnetwork Access Protocol

SNMP Simple Network Management Protocol

SNPA subnetwork point of attachment

SPF	OSPF intra-area route
SPE1	OSPF external route type 1
SPE2	OSPF external route type 2
SPIA	OSPF inter-area route type
SPID	service profile ID
SPX	Sequenced Packet Exchange
SQE	signal quality error
SRAM	static random access memory
SRB	source routing bridge
SRF	specifically routed frame
SRLY	SDLC relay
SRT	source routing transparent
SR-TB	source routing-transparent bridge
STA	static
STB	spanning tree bridge
STE	spanning tree explorer
STP	shielded twisted pair; spanning tree protocol
SVC	switched virtual circuit
TB	transparent bridge
TCN	topology change notification
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TEI	terminal point identifier
TFTP	Trivial File Transfer Protocol
TKR	token ring
TMO	timeout
TOS	type of service
TSF	transparent spanning frames
TTL	time to live
TTY	teletypewriter
TX	transmit
UA	unnumbered acknowledgment
UDP	User Datagram Protocol
UI	unnumbered information
UTP	unshielded twisted pair
VCC	Virtual Channel Connection
VINES	VIrtual NEtworking System
VIR	variable information rate
VL	virtual link
VNI	Vitrual Network Interface
VR	virtual route
WAN	wide area network
WRS	WAN restoral/reroute
X.25	packet-switched networks
X.251	X.25 physical layer
X.252	X.25 frame layer
X.253	X.25 packet layer
XID	exchange identification
XNS	Xerox Network Systems
XSUM	checksum
ZIP	AppleTalk Zone Information Protocol
ZIP2	AppleTalk Zone Information Protocol 2
ZIT	Zone Information Table

Glossary

This glossary includes terms and definitions from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.
- The ANSI/EIA Standard—440-A, *Fiber Optic Terminology* Copies may be purchased from the Electronic Industries Association, 2001 Pennsylvania Avenue, N.W., Washington, DC 20006. Definitions are identified by the symbol (E) after the definition.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.
- The *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.
- Internet Request for Comments: 1208, *Glossary of Networking Terms*
- Internet Request for Comments: 1392, *Internet Users' Glossary*
- The *Object-Oriented Interface Design: IBM Common User Access Guidelines*, Carmel, Indiana: Que, 1992.

The following cross-references are used in this glossary:

Contrast with: This refers to a term that has an opposed or substantively different meaning.

Synonym for: This indicates that the term has the same meaning as a preferred term, which is defined in its proper place in the glossary.

Synonymous with: This is a backward reference from a defined term to all other terms that have the same meaning.

See: This refers the reader to multiple-word terms that have the same last word.

See also: This refers the reader to terms that have a related, but not synonymous, meaning.

A

AAL. ATM Adaptation Layer, the layer that adapts user data to/from the ATM network by adding/removing headers and segmenting/reassembling the data into/from cells.

AAL-5. ATM Adaptation Layer 5, one of several standard AALs. AAL-5 was designed for data communications, and is used by LAN Emulation and Classical IP.

abstract syntax. A data specification that includes all distinctions that are needed in data transmissions, but that omits (abstracts) other details such as those that depend on specific computer architectures. See also *abstract syntax notation 1 (ASN.1)* and *basic encoding rules (BER)*.

abstract syntax notation 1 (ASN.1). The Open Systems Interconnection (OSI) method for abstract syntax specified in the following standards:

- ITU-T Recommendation X.208 (1988) | ISO/IEC 8824: 1990
- ITU-T Recommendation X.680 (1994) | ISO/IEC 8824-1: 1994

See also *basic encoding rules (BER)*.

ACCESS. In the Simple Network Management Protocol (SNMP), the clause in a Management Information Base (MIB) module that defines the minimum level of support that a managed node provides for an object.

acknowledgment. (1) The transmission, by a receiver, of acknowledge characters as an affirmative response to a sender. (T) (2) An indication that an item sent was received.

active. (1) Operational. (2) Pertaining to a node or device that is connected or is available for connection to another node or device.

active monitor. In a token-ring network, a function performed at any one time by one ring station that initiates the transmission of tokens and provides token error recovery facilities. Any active adapter on the ring has the ability to provide the active monitor function if the current active monitor fails.

address. In data communication, the unique code assigned to each device, workstation, or user connected to a network.

address mapping table (AMT). A table, maintained within the AppleTalk router, that provides a current mapping of node addresses to hardware addresses.

address mask. For internet subnetworking, a 32-bit mask used to identify the subnetwork address bits in the host portion of an IP address. Synonymous with *subnet mask* and *subnetwork mask*.

address resolution. (1) A method for mapping network-layer addresses to media-specific addresses. (2) See also *Address Resolution Protocol (ARP)* and *AppleTalk Address Resolution Protocol (AARP)*.

Address Resolution Protocol (ARP). (1) In the Internet suite of protocols, the protocol that dynamically maps an IP address to an address used by a supporting metropolitan or local area network such as Ethernet or token-ring. (2) See also *Reverse Address Resolution Protocol (RARP)*.

addressing. In data communication, the way in which a station selects the station to which it is to send data.

adjacent nodes. Two nodes connected together by at least one path that connects no other node. (T)

Administrative Domain. A collection of hosts and routers, and the interconnecting networks, managed by a single administrative authority.

Advanced Peer-to-Peer Networking (APPN). An extension to SNA featuring (a) greater distributed network control that avoids critical hierarchical dependencies, thereby isolating the effects of single points of failure; (b) dynamic exchange of network topology information to foster ease of connection, reconfiguration, and adaptive route selection; (c) dynamic definition of network resources; and (d) automated resource registration and directory lookup. APPN extends the LU 6.2 peer orientation for end-user services to network control and supports multiple LU types, including LU 2, LU 3, and LU 6.2.

Advanced Peer-to-Peer Networking (APPN) end node. A node that provides a broad range of end-user services and supports sessions between its local control point (CP) and the CP in an adjacent network node. It uses these sessions to dynamically register its resources with the adjacent CP (its network node server), to send and receive directory search requests, and to obtain management services. An APPN end node can also attach to a subarea network as a peripheral node or to other end nodes.

Advanced Peer-to-Peer Networking (APPN) network. A collection of interconnected network nodes and their client end nodes.

Advanced Peer-to-Peer Networking (APPN) network node. A node that offers a broad range of end-user services and that can provide the following:

- Distributed directory services, including registration of its domain resources to a central directory server
- Topology database exchanges with other APPN network nodes, enabling network nodes throughout the network to select optimal routes for LU-LU sessions based on requested classes of service
- Session services for its local LUs and client end nodes
- Intermediate routing services within an APPN network

Advanced Peer-to-Peer Networking (APPN) node. An APPN network node or an APPN end node.

agent. A system that assumes an agent role.

alert. A message sent to a management services focal point in a network to identify a problem or an impending problem.

all-stations address. In communications, synonym for *broadcast address*.

American National Standards Institute (ANSI). An organization consisting of producers, consumers, and general interest groups, that establishes the procedures by which accredited organizations create and maintain voluntary industry standards in the United States. (A)

analog. (1) Pertaining to data consisting of continuously variable physical quantities. (A) (2) Contrast with *digital*.

AppleTalk. A network protocol developed by Apple Computer, Inc. This protocol is used to interconnect network devices, which can be a mixture of Apple and non-Apple products.

AppleTalk Address Resolution Protocol (AARP). In AppleTalk networks, a protocol that (a) translates AppleTalk node addresses into hardware addresses and (b) reconciles addressing discrepancies in networks that support more than one set of protocols.

AppleTalk Transaction Protocol (ATP). In AppleTalk networks, a protocol that provides client/server request and response functions for hosts accessing the Zone Information Protocol (ZIP) for zone information.

APPN network. See *Advanced Peer-to-Peer Networking (APPN) network*.

APPN network node. See *Advanced Peer-to-Peer Networking (APPN) network node*.

arbitrary MAC addressing (AMA). In DECnet architecture, an addressing scheme used by DECnet Phase IV-Prime that supports universally administered addresses and locally administered addresses.

area. In Internet and DECnet routing protocols, a subset of a network or gateway grouped together by definition of the network administrator. Each area is self-contained; knowledge of an area's topology remains hidden from other areas.

asynchronous (ASYNC). Pertaining to two or more processes that do not depend upon the occurrence of specific events such as common timing signals. (T)

ATM. Asynchonous Transfer Mode, a connection-oriented, high-speed networking technology based on cell switching.

ATMARP. ARP in Classical IP.

attachment unit interface (AUI). In a local area network, the interface between the medium attachment unit and the data terminal equipment within a data station. (I) (A)

authentication failure. In the Simple Network Management Protocol (SNMP), a trap that may be generated by an authentication entity when a requesting client is not a member of the SNMP community.

autonomous system. In TCP/IP, a group of networks and routers under one administrative authority. These networks and routers cooperate closely to propagate network reachability (and routing) information among themselves using an interior gateway protocol of their choice.

autonomous system number. In TCP/IP, a number assigned to an autonomous system by the same central authority that also assigns IP addresses. The autonomous system number makes it possible for automated routing algorithms to distinguish autonomous systems.

B

backbone. (1) In a local area network multiple-bridge ring configuration, a high-speed link to which the rings are connected by means of bridges or routers. A backbone may be configured as a bus or as a ring. (2) In a wide area network, a high-speed link to which nodes or data switching exchanges (DSEs) are connected.

backbone network. A central network to which smaller networks, normally of lower speed, connect. The backbone network usually has a much higher capacity than the networks it helps interconnect or is a wide-area network (WAN) such as a public packet-switched datagram network.

backbone router. (1) A router used to transmit data between areas. (2) One in a series of routers that is used to interconnect networks into a larger internet.

Bandwidth. The bandwidth of an optical link designates the information-carrying capacity of the link and is related to the maximum bit rate that a fiber link can support.

basic transmission unit (BTU). In SNA, the unit of data and control information passed between path control components. A BTU can consist of one or more path information units (PIUs).

baud. In asynchronous transmission, the unit of modulation rate corresponding to one unit interval per second; that is, if the duration of the unit interval is 20 milliseconds, the modulation rate is 50 baud. (A)

bootstrap. (1) A sequence of instructions whose execution causes additional instructions to be loaded and executed until the complete computer program is in storage. (T) (2) A technique or device designed to bring itself into a desired state by means of its own action, for example, a machine routine whose first few instructions are sufficient to bring the rest of itself into the computer from an input device. (A)

Border Gateway Protocol (BGP). An Internet Protocol (IP) routing protocol used between domains and autonomous systems.

border router. In Internet communications, a router, positioned at the edge of an autonomous system, that communicates with a router that is positioned at the edge of a different autonomous system.

bridge. A functional unit that interconnects multiple LANs (locally or remotely) that use the same logical link control protocol but that can use different medium access control protocols. A bridge forwards a frame to another bridge based on the medium access control (MAC) address.

bridge identifier. An 8-byte field, used in a spanning tree protocol, composed of the MAC address of the port with the lowest port identifier and a user-defined value.

bridging. In LANs, the forwarding of a frame from one LAN segment to another. The destination is specified by the medium access control (MAC) sublayer address encoded in the destination address field of the frame header.

broadcast. (1) Transmission of the same data to all destinations. (T) (2) Simultaneous transmission of data

to more than one destination. (3) Contrast with *multicast*.

broadcast address. In communications, a station address (eight 1's) reserved as an address common to all stations on a link. Synonymous with *all-stations address*.

C

cache. (1) A special-purpose buffer storage, smaller and faster than main storage, used to hold a copy of instructions and data obtained from main storage and likely to be needed next by the processor. (T) (2) A buffer storage that contains frequently accessed instructions and data; it is used to reduce access time. (3) An optional part of the directory database in network nodes where frequently used directory information may be stored to speed directory searches. (4) To place, hide, or store in a cache.

call request packet. (1) A call supervision packet that a data terminal equipment (DTE) transmits to ask that a connection for a call be established throughout the network. (2) In X.25 communications, a call supervision packet transmitted by a DTE to ask for a call establishment through the network.

canonical address. In LANs, the IEEE 802.1 format for the transmission of medium access control (MAC) addresses for token-ring and Ethernet adapters. In canonical format, the least significant (rightmost) bit of each address byte is transmitted first. Contrast with *noncanonical address*.

carrier. An electric or electromagnetic wave or pulse train that may be varied by a signal bearing information to be transmitted over a communication system. (T)

carrier detect. Synonym for *received line signal detector (RLSD)*.

carrier sense. In a local area network, an ongoing activity of a data station to detect whether another station is transmitting. (T)

carrier sense multiple access with collision detection (CSMA/CD). A protocol that requires carrier sense and in which a transmitting data station that detects another signal while transmitting, stops sending, sends a jam signal, and then waits for a variable time before trying again. (T) (A)

CCITT. International Telegraph and Telephone Consultative Committee. This was an organization of the International Telecommunication Union (ITU). On 1 March 1993 the ITU was reorganized, and responsibilities for standardization were placed in a subordinate organization named the Telecommunication Standardization Sector of the Telecommunication Union (ITU-TS). “CCITT” continues to be used for recommendations that were approved before the reorganization.

channel. (1) A path along which signals can be sent, for example, data channel, output channel. (A) (2) A functional unit, controlled by the processor, that handles the transfer of data between processor storage and local peripheral equipment.

channel service unit (CSU). A unit that provides the interface to a digital network. The CSU provides line conditioning (or equalization) functions, which keep the signal's performance consistent across the channel bandwidth; signal reshaping, which constitutes the binary pulse stream; and loopback testing, which includes the transmission of test signals between the CSU and the network carrier's office channel unit. See also *data service unit (DSU)*.

checksum. (1) The sum of a group of data associated with the group and used for checking purposes. (T) (2) In error detection, a function of all bits in a block. If the written and calculated sums do not agree, an error is indicated. (3) On a diskette, data written in a sector for error detection purposes; a calculated checksum that does not match the checksum of data written in the sector indicates a bad sector. The data are either numeric or other character strings regarded as numeric for the purpose of calculating the checksum.

circuit switching. (1) A process that, on demand, connects two or more data terminal equipment (DTEs) and permits the exclusive use of a data circuit between them until the connection is released. (I) (A) (2) Synonymous with *line switching*.

class A network. In Internet communications, a network in which the high-order (most significant) bit of the IP address is set to 0 and the host ID occupies the three low-order octets.

class B network. In Internet communications, a network in which the two high-order (most significant and next-to-most significant) bits of the IP address are set to 1 and 0, respectively, and the host ID occupies the two low-order octets.

class of service (COS). A set of characteristics (such as route security, transmission priority, and bandwidth) used to construct a route between session partners. The class of service is derived from a mode name specified by the initiator of a session.

client. (1) A functional unit that receives shared services from a server. (T) (2) A user.

client/server. In communications, the model of interaction in distributed data processing in which a program at one site sends a request to a program at

another site and awaits a response. The requesting program is called a client; the answering program is called a server.

clocking. (1) In binary synchronous communication, the use of clock pulses to control synchronization of data and control characters. (2) A method of controlling the number of data bits sent on a telecommunication line in a given time.

collision. An unwanted condition that results from concurrent transmissions on a channel. (T)

collision detection. In carrier sense multiple access with collision detection (CSMA/CD), a signal indicating that two or more stations are transmitting simultaneously.

Committed information rate. The maximum amount of data in bits that the network agrees to deliver.

community. In the Simple Network Management Protocol (SNMP), an administrative relationship between entities.

community name. In the Simple Network Management Protocol (SNMP), a string of octets identifying a community.

compression. (1) The process of eliminating gaps, empty fields, redundancies, and unnecessary data to shorten the length of records or blocks. (2) Any encoding to reduce the number of bits used to represent a given message or record.

configuration. (1) The manner in which the hardware and software of an information processing system are organized and interconnected. (T) (2) The devices and programs that make up a system, subsystem, or network.

configuration database (CDB). A database that stores the configuration parameters of one or several devices. It is prepared and updated using the configuration program.

configuration file. A file that specifies the characteristics of a system device or network.

configuration parameter. A variable in a configuration definition, the values of which can characterize the relationship of a product to other products in the same network or can define characteristics of the product itself.

configuration report server (CRS). In the IBM Token-Ring Network Bridge Program, the server that accepts commands from the LAN Network Manager (LNM) to get station information, set station parameters, and remove stations on its ring. This server also collects and forwards configuration reports generated by stations on its ring. The configuration reports include the new active monitor reports and the nearest active upstream neighbor (NAUN) reports.

congestion. See *network congestion*.

connection. In data communication, an association established between functional units for conveying information. (I) (A)

control point (CP). (1) A component of an APPN or LEN node that manages the resources of that node. In an APPN node, the CP is capable of engaging in CP-CP sessions with other APPN nodes. In an APPN network node, the CP also provides services to adjacent end nodes in the APPN network. (2) A component of a node that manages resources of that node and optionally provides services to other nodes in the network. Examples are a system services control point (SSCP) in a type 5 subarea node, a network node control point (NNCP) in an APPN network node, and an end node control point (ENCP) in an APPN or LEN end node. An SSCP and an NNCP can provide services to other nodes.

control point management services (CPMS). A component of a control point, consisting of management services function sets, that provides facilities to assist in performing problem management, performance and accounting management, change management, and configuration management. Capabilities provided by the CPMS include sending requests to physical unit management services (PUMS) to test system resources, collecting statistical information (for example, error and performance data) from PUMS about the system resources, and analyzing and presenting test results and statistical information collected about the system resources. Analysis and presentation responsibilities for problem determination and performance monitoring can be distributed among multiple CPMSs.

control point management services unit (CP-MSU). The message unit that contains management services data and flows between management services function sets. This message unit is in general data stream (GDS) format. See also *management services unit (MSU)* and *network management vector transport (NMVT)*.

D

D-bit. Delivery-confirmation bit. In X.25 communications, the bit in a data packet or call-request packet that is set to 1 if end-to-end acknowledgment (delivery confirmation) is required from the recipient.

daemon. A program that runs unattended to perform a standard service. Some daemons are triggered

automatically to perform their task; others operate periodically.

data carrier detect (DCD). Synonym for *received line signal detector (RLSD)*.

data circuit. (1) A pair of associated transmit and receive channels that provide a means of two-way data communication. (I) (2) In SNA, synonym for *link connection*. (3) See also *physical circuit* and *virtual circuit*.

Notes:

1. Between data switching exchanges, the data circuit may include data circuit-terminating equipment (DCE), depending on the type of interface used at the data switching exchange.
2. Between a data station and a data switching exchange or data concentrator, the data circuit includes the data circuit-terminating equipment at the data station end, and may include equipment similar to a DCE at the data switching exchange or data concentrator location.

data circuit-terminating equipment (DCE). In a data station, the equipment that provides the signal conversion and coding between the data terminal equipment (DTE) and the line. (I)

Notes:

1. The DCE may be separate equipment or an integral part of the DTE or of the intermediate equipment.
2. A DCE may perform other functions that are usually performed at the network end of the line.

data link connection identifier (DLCI). The numeric identifier of a frame-relay subport or PVC segment in a frame-relay network. Each subport in a single frame-relay port has a unique DLCI. The following table, excerpted from the American National Standards Institute (ANSI) Standard T1.618 and the International Telegraph and Telephone Consultative Committee (ITU-T/CCITT) Standard Q.922, indicates the functions associated with certain DLCI values:

DLCI Values	Function
0	in-channel signaling
1–15	reserved
16–991	assigned using frame-relay connection procedures
992–1007	layer 2 management of frame-relay bearer service
1008–1022	reserved
1023	in-channel layer management

data link control (DLC). A set of rules used by nodes on a data link (such as an SDLC link or a token ring) to accomplish an orderly exchange of information.

data link control (DLC) layer. In SNA, the layer that consists of the link stations that schedule data transfer over a link between two nodes and perform error control for the link. Examples of data link control are SDLC for serial-by-bit link connection and data link control for the System/370 channel.

Note: The DLC layer is usually independent of the physical transport mechanism and ensures the integrity of data that reaches the higher layers.

data link layer. In the Open Systems Interconnection reference model, the layer that provides services to transfer data between entities in the network layer over a communication link. The data link layer detects and possibly corrects errors that may occur in the physical layer. (T)

data link level. (1) In the hierarchical structure of a data station, the conceptual level of control or processing logic between high level logic and the data link that maintains control of the data link. The data link level performs such functions as inserting transmit bits and deleting receive bits; interpreting address and control fields; generating, transmitting, and interpreting commands and responses; and computing and interpreting frame check sequences. See also *packet level* and *physical level*. (2) In X.25 communications, synonym for *frame level*.

data link switching (DLSw). A method of transporting network protocols that use IEEE 802.2 logical link control (LLC) type 2. SNA and NetBIOS are examples of protocols that use LLC type 2. See also *encapsulation* and *spoofing*.

data packet. In X.25 communications, a packet used for the transmission of user data on a virtual circuit at the DTE/DCE interface.

data service unit (DSU). A device that provides a digital data service interface directly to the data terminal equipment. The DSU provides loop equalization, remote and local testing capabilities, and a standard EIA/CCITT interface.

data set ready (DSR). Synonym for *DCE ready*.

data switching exchange (DSE). The equipment installed at a single location to provide switching functions, such as circuit switching, message switching, and packet switching. (I)

data terminal equipment (DTE). That part of a data station that serves as a data source, data sink, or both. (I) (A)

data terminal ready (DTR). A signal to the modem used with the EIA 232 protocol.

data transfer rate. The average number of bits, characters, or blocks per unit time passing between corresponding equipment in a data transmission system. (I)

Notes:

1. The rate is expressed in bits, characters, or blocks per second, minute, or hour.
2. Corresponding equipment should be indicated; for example, modems, intermediate equipment, or source and sink.

datagram. (1) In packet switching, a self-contained packet, independent of other packets, that carries information sufficient for routing from the originating data terminal equipment (DTE) to the destination DTE without relying on earlier exchanges between the DTEs and the network. (I) (2) In TCP/IP, the basic unit of information passed across the Internet environment. A datagram contains a source and destination address along with the data. An Internet Protocol (IP) datagram consists of an IP header followed by the transport layer data. (3) See also *packet* and *segment*.

Datagram Delivery Protocol (DDP). In AppleTalk networks, a protocol that provides network connectivity by means of connectionless socket-to-socket delivery service on the internet layer.

DCE ready. In the EIA 232 standard, a signal that indicates to the data terminal equipment (DTE) that the local data circuit-terminating equipment (DCE) is connected to the communication channel and is ready to send data. Synonymous with *data set ready (DSR)*.

DECnet. A network architecture that defines the operation of a family of software modules, databases, and hardware components typically used to tie Digital Equipment Corporation systems together for resource sharing, distributed computation, or remote system configuration. DECnet network implementations follow the Digital Network Architecture (DNA) model.

default. Pertaining to an attribute, condition, value, or option that is assumed when none is explicitly specified. (I)

dependent LU requester (DLUR). An APPN end node or an APPN network node that owns dependent LUs, but requests that a dependent LU server provide the SSCP services for those dependent LUs.

designated router. A router that informs end nodes of the existence and identity of other routers. The selection of the designated router is based upon the router with the highest priority. When several routers share the highest priority, the router with the highest station address is selected.

destination node. The node to which a request or data is sent.

destination port. The 8-port asynchronous adapter that serves as a connection point with a serial service.

destination service access point (DSAP). In SNA and TCP/IP, a logical address that allows a system to route data from a remote device to the appropriate communications support. Contrast with *source service access point (SSAP)*.

device. A mechanical, electrical, or electronic contrivance with a specific purpose.

digital. (1) Pertaining to data that consist of digits. (T) (2) Pertaining to data in the form of digits. (A) (3) Contrast with *analog*.

Digital Network Architecture (DNA). The model for all DECnet hardware and software implementations.

direct memory access (DMA). The system facility that allows a device on the Micro Channel bus to get direct access to the system or bus memory without the intervention of the system processor.

directory. A table of identifiers and references to the corresponding items of data. (I) (A)

directory service (DS). An application service element that translates the symbolic names used by application processes into the complete network addresses used in an OSI environment. (T)

directory services (DS). A control point component of an APPN node that maintains knowledge of the location of network resources.

disable. To make nonfunctional.

disabled. (1) Pertaining to a state of a processing unit that prevents the occurrence of certain types of interruptions. (2) Pertaining to the state in which a transmission control unit or audio response unit cannot accept incoming calls on a line.

domain. (1) That part of a computer network in which the data processing resources are under common control. (T) (2) In Open Systems Interconnection (OSI), a part of a distributed system or a set of managed objects to which a common policy applies. (3) See *Administrative Domain* and *domain name*.

domain name. In the Internet suite of protocols, a name of a host system. A domain name consists of a sequence of subnames separated by a delimiter character. For example, if the fully qualified domain name (FQDN) of a host system is `ralvm7.vnet.ibm.com`, each of the following is a domain name:

- ralvm7.vnet.ibm.com
- vnet.ibm.com
- ibm.com

domain name server. In the Internet suite of protocols, a server program that supplies name-to-address translation by mapping domain names to IP addresses. Synonymous with *name server*.

Domain Name System (DNS). In the Internet suite of protocols, the distributed database system used to map domain names to IP addresses.

dotted decimal notation. The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 with periods (dots) separating them. It is used to represent IP addresses.

dump. (1) Data that has been dumped. (T) (2) To copy the contents of all or part of virtual storage for the purpose of collecting error information.

dynamic reconfiguration (DR). The process of changing the network configuration (peripheral PUs and LUs) without regenerating complete configuration tables or deactivating the affected major node.

Dynamic Routing. Routing using learned routes rather than routes statically configured at initialization.

E

echo. In data communication, a reflected signal on a communications channel. For example, on a communications terminal, each signal is displayed twice, once when entered at the local terminal and again when returned over the communications link. This allows the signals to be checked for accuracy.

EIA 232. In data communication, a specification of the Electronic Industries Association (EIA) that defines the interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE), using serial binary data interchange.

Electronic Industries Association (EIA). An organization of electronics manufacturers that advances the technological growth of the industry, represents the views of its members, and develops industry standards.

EIA unit. A unit of measure, established by the Electronic Industries Association, equal to 44.45 millimeters (1.75 inches).

encapsulation. (1) In communications, a technique used by layered protocols by which a layer adds control information to the protocol data unit (PDU) from the layer it supports. In this respect, the layer encapsulates the data from the supported layer. In the Internet suite of protocols, for example, a packet would contain control information from the physical layer, followed by control information from the network layer, followed by the application protocol data. (2) See also *data link switching*.

encode. To convert data by the use of a code in such a manner that reconversion to the original form is possible. (T)

end node (EN). (1) See *Advanced Peer-to-Peer Networking (APPN) end node* and *low-entry networking (LEN) end node*. (2) In communications, a node that is frequently attached to a single data link and cannot perform intermediate routing functions.

entry point (EP). In SNA, a type 2.0, type 2.1, type 4, or type 5 node that provides distributed network management support. It sends network management data about itself and the resources it controls to a focal point for centralized processing, and it receives and executes focal-point initiated commands to manage and control its resources.

Ethernet. A 10-Mbps baseband local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and delayed retransmission. Ethernet uses carrier sense multiple access with collision detection (CSMA/CD).

exception. An abnormal condition such as an I/O error encountered in processing a data set or a file.

exception response (ER). In SNA, a protocol requested in the form-of-response-requested field of a request header that directs the receiver to return a response only if the request is unacceptable as received or cannot be processed; that is, a negative response, but not a positive response, can be returned. Contrast with *definite response* and *no response*.

exchange identification (XID). A specific type of basic link unit that is used to convey node and link characteristics between adjacent nodes. XIDs are exchanged between link stations before and during link activation to establish and negotiate link and node characteristics, and after link activation to communicate changes in these characteristics.

explicit route (ER). In SNA, a series of one or more transmission groups that connect two subarea nodes. An explicit route is identified by an origin subarea address, a destination subarea address, an explicit route number, and a reverse explicit route number. Contrast with *virtual route (VR)*.

explorer frame. See *explorer packet*.

explorer packet. In LANs, a packet that is generated by the source host and that traverses the entire source

routing part of a LAN, gathering information on the possible paths available to the host.

exterior gateway. In Internet communications, a gateway on one autonomous system that communicates with another autonomous system. Contrast with *interior gateway*.

Exterior Gateway Protocol (EGP). In the Internet suite of protocols, a protocol, used between domains and autonomous systems, that enables network reachability information to be advertised and exchanged. IP network addresses in one autonomous system are advertised to another autonomous system by means of EGP-participating routers. An example of an EGP is the Border Gateway Protocol (BGP). Contrast with Interior Gateway Protocol (IGP).

F

fax. Hardcopy received from a facsimile machine. Synonymous with ***telecopy.***

File Transfer Protocol (FTP). In the Internet suite of protocols, an application layer protocol that uses TCP and Telnet services to transfer bulk-data files between machines or hosts.

flash memory. A data storage device that is programmable, erasable, and does not require continuous power. The chief advantage of flash memory over other programmable and erasable data storage devices is that it can be reprogrammed without being removed from the circuit board.

flow control. (1) In SNA, the process of managing the rate at which data traffic passes between components of the network. The purpose of flow control is to optimize the rate of flow of message units with minimum congestion in the network; that is, to neither overflow the buffers at the receiver or at intermediate routing nodes, nor leave the receiver waiting for more message units. (2) See also *pacing*.

fragment. See *fragmentation*.

fragmentation. (1) The process of dividing a datagram into smaller parts, or fragments, to match the capabilities of the physical medium over which it is to be transmitted. (2) See also *segmenting*.

frame. (1) In Open Systems Interconnection architecture, a data structure pertaining to a particular area of knowledge and consisting of slots that can accept the values of specific attributes and from which inferences can be drawn by appropriate procedural attachments. (T) (2) The unit of transmission in some local area networks, including the IBM Token-Ring Network. It includes delimiters, control characters, information, and checking characters. (3) In SDLC, the vehicle for every command, every response, and all information that is transmitted using SDLC procedures.

frame level. Synonymous with *data link level*. See *link level*.

frame relay. (1) An interface standard describing the boundary between a user's equipment and a fast-packet network. In frame-relay systems, flawed frames are discarded; recovery comes end-to-end rather than hop-by-hop. (2) A technique derived from the integrated services digital network (ISDN) D channel standard. It assumes that connections are reliable and dispenses with the overhead of error detection and control within the network.

front-end processor. A processor such as the IBM 3745 or 3174, that relieves a main frame from the communication control tasks.

G

gateway. (1) A functional unit that interconnects two computer networks with different network architectures. A gateway connects networks or systems of different architectures. A bridge interconnects networks or systems with the same or similar architectures. (T) (2) In the IBM Token-Ring Network, a device and its associated software that connect a local area network to another local area network or a host that uses different logical link protocols. (3) In TCP/IP, synonym for *router*.

general data stream (GDS). The data stream used for conversations in LU 6.2 sessions.

general data stream (GDS) variable. A type of RU substructure that is preceded by an identifier and a length field and includes either application data, user control data, or SNA-defined control data.

H

header. (1) System-defined control information that precedes user data. (2) The portion of a message that contains control information for the message such as one or more destination fields, name of the originating station, input sequence number, character string indicating the type of message, and priority level for the message.

heap memory. The amount of RAM used to dynamically allocate data structures.

Hello. A protocol used by a group of cooperating, trusting routers to allow them to discover minimal delay routes.

hello message. (1) A message sent periodically to establish and test reachability between routers or between routers and hosts. (2) In the Internet suite of protocols, a message defined by the Hello protocol as an Interior Gateway Protocol (IGP).

heuristic. Pertaining to exploratory methods of problem solving in which solutions are discovered by evaluation of the progress made toward the final result.

high-level data link control (HDLC). In data communication, the use of a specified series of bits to control data links in accordance with the International Standards for HDLC: ISO 3309 Frame Structure and ISO 4335 Elements of Procedures.

high-performance routing (HPR). An addition to the Advanced Peer-to-Peer Networking (APPN) architecture that enhances data routing performance and reliability, especially when using high-speed links.

hop. (1) In APPN, a portion of a route that has no intermediate nodes. It consists of only a single transmission group connecting adjacent nodes. (2) To the routing layer, the logical distance between two nodes in a network.

hop count. (1) A metric or measure of distance between two points. (2) In Internet communications, the number of routers that a datagram passes through on its way to its destination. (3) In SNA, a measure of the number of links to be traversed in a path to a destination.

host. In the Internet suite of protocols, an end system. The end system can be any workstation; it does not have to be a mainframe.

hub (intelligent). A wiring concentrator, such as the IBM 8260, that provides bridging and routing functions for LANs with different cables and protocols.

hysteresis. The amount the temperature must change past the set alert threshold before the alert condition is cleared.

I

I-frame. Information frame.

information (I) frame. A frame in I format used for numbered information transfer.

input/output channel. In a data processing system, a functional unit that handles transfer of data between internal and peripheral equipment. (I) (A)

Integrated Digital Network Exchange (IDNX). A processor integrating voice, data, and image applications. It also manages the transmission resources, and connects to multiplexers and network management support systems. It allows integration of equipment from different vendors.

integrated services digital network (ISDN). A digital end-to-end telecommunication network that supports multiple services including, but not limited to, voice and data.

Note: ISDNs are used in public and private network architectures.

interface. (1) A shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics, as appropriate. The concept includes the specification of the connection of two devices having different functions. (T) (2) Hardware, software, or both, that links systems, programs, or devices.

interior gateway. In Internet communications, a gateway that communicates only with its own autonomous system. Contrast with *exterior gateway*.

Interior Gateway Protocol (IGP). In the Internet suite of protocols, a protocol used to propagate network reachability and routing information within an autonomous system. Examples of IGPs are Routing Information Protocol (RIP) and Open Shortest Path First (OSPF).

intermediate node. A node that is at the end of more than one branch. (T)

intermediate session routing (ISR). A type of routing function within an APPN network node that provides session-level flow control and outage reporting for all sessions that pass through the node but whose end points are elsewhere.

International Organization for Standardization (ISO). An organization of national standards bodies from various countries established to promote development of standards to facilitate international exchange of goods and services, and develop cooperation in intellectual, scientific, technological, and economic activity.

International Telecommunication Union (ITU). The specialized telecommunication agency of the United Nations, established to provide standardized communication procedures and practices, including frequency allocation and radio regulations worldwide.

internet. A collection of networks interconnected by a set of routers that allow them to function as a single, large network. See also *Internet*.

Internet. The internet administered by the Internet Architecture Board (IAB), consisting of large national backbone networks and many regional and campus

networks all over the world. The Internet uses the Internet suite of protocols.

Internet address. See *IP address*.

Internet Architecture Board (IAB). The technical body that oversees the development of the Internet suite of protocols that are known as TCP/IP.

Internet Control Message Protocol (ICMP). The protocol used to handle errors and control messages in the Internet Protocol (IP) layer. Reports of problems and incorrect datagram destinations are returned to the original datagram source. ICMP is part of the Internet Protocol.

Internet Control Protocol (ICP). The VIrtual NEtworking System (VINES) protocol that provides exception notifications, metric notifications, and PING support. See also *RouTing update Protocol (RTP)*.

Internet Engineering Task Force (IETF). The task force of the Internet Architecture Board (IAB) that is responsible for solving the short-term engineering needs of the Internet.

Internetwork Packet Exchange (IPX). (1) The network protocol used to connect Novell's servers, or any workstation or router that implements IPX, with other workstations. Although similar to the Internet Protocol (IP), IPX uses different packet formats and terminology. (2) See also *Xerox Network Systems (XNS)*.

Internet Protocol (IP). A connectionless protocol that routes data through a network or interconnected networks. IP acts as an intermediary between the higher protocol layers and the physical network. However, this protocol does not provide error recovery and flow control and does not guarantee the reliability of the physical network.

interoperability. The capability to communicate, execute programs, or transfer data among various functional units in a way that requires the user to have little or no knowledge of the unique characteristics of those units. (T)

intra-area routing. In Internet communications, the routing of data within an area.

Inverse Address Resolution Protocol (InARP). In the Internet suite of protocols, the protocol used for locating a protocol address through the known hardware address. In a frame-relay context, the data link connection identifier (DLCI) is synonymous with the known hardware address.

IP address. The 32-bit address defined by the Internet Protocol, standard 5, Request for Comments (RFC) 791. It is usually represented in dotted decimal notation.

IP datagram. In the Internet suite of protocols, the fundamental unit of information transmitted through an internet. It contains source and destination addresses, user data, and control information such as the length of the datagram, the header checksum, and flags indicating whether the datagram can be or has been fragmented.

IP router. A device in an IP internet that is responsible for making decisions about the paths over which network traffic will flow. Routing protocols are used to gain information about the network and to determine the best route over which the datagram should be forwarded toward the final destination. The datagrams are routed based on IP destination addresses.

IPXWAN. A Novell protocol that is used to exchange router-to-router information before exchanging standard Internetwork Packet Exchange (IPX) routing information and traffic over wide area networks (WANs).

L

LAN bridge server (LBS). In the IBM Token-Ring Network Bridge Program, the server that keeps statistical information about frames forwarded between two or more rings (through a bridge). The LBS sends these statistics to the appropriate LAN managers through the LAN reporting mechanism (LRM).

LAN Emulation (LE). An ATM Forum standard that supports legacy LAN applications over ATM networks.

LAN Emulation Client (LEC). A LAN Emulation component that represents users of the Emulated LAN.

LAN Emulation Configuration Server (LECS). A LAN Emulation Service component that centralizes and disseminates configuration data.

LAN Emulation Server (LES). A LAN Emulation Service component that resolves LAN Destinations to ATM Addresses.

LAN Network Manager (LNM). An IBM licensed program that enables a user to manage and monitor LAN resources from a central workstation.

LAN segment. (1) Any portion of a LAN (for example, a bus or ring) that can operate independently, but that is connected to other parts of the network by means of bridges. (2) A ring or bus network without bridges.

layer. (1) In network architecture, a group of services that is complete from a conceptual point of view, that is one out of a set of hierarchically arranged groups, and that extends across all systems that conform to the network architecture. (T) (2) In the Open Systems Interconnection reference model, one of seven

conceptually complete, hierarchically arranged groups of services, functions, and protocols, that extend across all open systems. (T) (3) In SNA, a grouping of related functions that are logically separate from the functions in other groups. Implementation of the functions in one layer can be changed without affecting functions in other layers.

LE. LAN Emulation. An ATM Forum standard that supports legacy LAN applications over ATM networks.

LEC. LAN Emulation Client. A LAN Emulation component that represents users of the Emulated LAN.

LECS. LAN Emulation Configuration Server. A LAN Emulation Service component that centralizes and disseminates configuration data.

LES. LAN Emulation Server. A LAN Emulation Service component that resolves LAN Destinations to ATM Addresses.

line switching. Synonym for *circuit switching*.

link. The combination of the link connection (the transmission medium) and two link stations, one at each end of the link connection. A link connection can be shared among multiple links in a multipoint or token-ring configuration.

link access protocol balanced (LAPB). A protocol used for accessing an X.25 network at the link level. LAPB is a duplex, asynchronous, symmetric protocol, used in point-to-point communication.

link-attached. (1) Pertaining to devices that are connected to a controlling unit by a data link. (2) Contrast with *channel-attached*. (3) Synonymous with *remote*.

link connection. (1) The physical equipment providing two-way communication between one link station and one or more other link stations; for example, a telecommunication line and data circuit-terminating equipment (DCE). (2) In SNA, synonymous with *data circuit*.

link level. (1) A part of Recommendation X.25 that defines the link protocol used to get data into and out of the network across the full-duplex link connecting the subscriber's machine to the network node. LAP and LAPB are the link access protocols recommended by the CCITT. (2) See *data link level*.

link-state. In routing protocols, the advertised information about the usable interfaces and reachable neighbors of a router or network. The protocol's topological database is formed from the collected link-state advertisements.

link station. (1) The hardware and software components within a node representing a connection to an adjacent node over a specific link. For example, if node A is the primary end of a multipoint line that connects to three adjacent nodes, node A will have three link stations representing the connections to the adjacent nodes. (2) See also *adjacent link station (ALS)*.

local. (1) Pertaining to a device accessed directly without use of a telecommunication line. (2) Contrast with *remote*. (3) Synonym for *channel-attached*.

local area network (LAN). (1) A computer network located on a user's premises within a limited geographical area. Communication within a local area network is not subject to external regulations; however, communication across the LAN boundary may be subject to some form of regulation. (T) (2) A network in which a set of devices are connected to one another for communication and that can be connected to a larger network. (3) See also *Ethernet* and *token ring*. (4) Contrast with *metropolitan area network (MAN)* and *wide area network (WAN)*.

local bridging. A function of a bridge program that allows a single bridge to connect multiple LAN segments without using a telecommunication link. Contrast with *remote bridging*.

local management interface (LMI). See *local management interface (LMI) protocol*.

local management interface (LMI) protocol. In NCP, a set of frame-relay network management procedures and messages used by adjacent frame-relay nodes to exchange line status information over DLCI X'00'. NCP supports both the American National Standards Institute (ANSI) and International Telegraph and Telephone Consultative Committee (ITU-T/CCITT) versions of LMI protocol. These standards refer to LMI protocol as *link integrity verification tests (LIVT)*.

locally administered address. In a local area network, an adapter address that the user can assign to override the universally administered address. Contrast with *universally administered address*.

logical channel. In packet mode operation, a sending channel and a receiving channel that together are used to send and receive data over a data link at the same time. Several logical channels can be established on the same data link by interleaving the transmission of packets.

logical link. A pair of link stations, one in each of two adjacent nodes, and their underlying link connection, providing a single link-layer connection between the two nodes. Multiple logical links can be distinguished while they share the use of the same physical media

connecting two nodes. Examples are 802.2 logical links used on local area network (LAN) facilities and LAP E logical links on the same point-to-point physical link between two nodes. The term logical link also includes the multiple X.25 logical channels that share the use of the access link from a DTE to an X.25 network.

logical link control (LLC). The data link control (DLC) LAN sublayer that provides two types of DLC operation for the orderly exchange of information. The first type is connectionless service, which allows information to be sent and received without establishing a link. The LLC sublayer does not perform error recovery or flow control for connectionless service. The second type is connection-oriented service, which requires establishing a link prior to the exchange of information. Connection-oriented service provides sequenced information transfer, flow control, and error recovery.

logical link control (LLC) protocol. In a local area network, the protocol that governs the exchange of transmission frames between data stations independently of how the transmission medium is shared. (T) The LLC protocol was developed by the IEEE 802 committee and is common to all LAN standards.

logical link control (LLC) protocol data unit. A unit of information exchanged between link stations in different nodes. The LLC protocol data unit contains a destination service access point (DSAP), a source service access point (SSAP), a control field, and user data.

logical unit (LU). A type of network accessible unit that enables users to gain access to network resources and communicate with each other.

loopback test. A test in which signals from a tester are looped at a modem or other network element back to the tester for measurements that determine or verify the quality of the communications path.

low-entry networking (LEN). A capability of nodes to attach directly to one another using basic peer-to-peer protocols to support multiple and parallel sessions between logical units.

low-entry networking (LEN) end node. A LEN node receiving network services from an adjacent APPN network node.

low-entry networking (LEN) node. A node that provides a range of end-user services, attaches directly to other nodes using peer protocols, and derives network services implicitly from an adjacent APPN network node, that is, without the direct use of CP-CP sessions.

M

Management Information Base (MIB). (1) A collection of objects that can be accessed by means of a network management protocol. (2) A definition for management information that specifies the information available from a host or gateway and the operations allowed. (3) In OSI, the conceptual repository of management information within an open system.

management station. In Internet communications, the system responsible for managing all, or a portion of, a network. The management station communicates with network management agents that reside in the managed node by means of a network management protocol, such as the Simple Network Management Protocol (SNMP).

mapping. The process of converting data that is transmitted in one format by the sender into the data format that can be accepted by the receiver.

mask. (1) A pattern of characters used to control retention or elimination of portions of another pattern of characters. (I) (A) (2) To use a pattern of characters to control retention or elimination of portions of another pattern of characters. (I) (A)

maximum transmission unit (MTU). In LANs, the largest possible unit of data that can be sent on a given physical medium in a single frame. For example, the MTU for Ethernet is 1500 bytes.

medium access control (MAC). In LANs, the sublayer of the data link control layer that supports medium-dependent functions and uses the services of the physical layer to provide services to the logical link control (LLC) sublayer. The MAC sublayer includes the method of determining when a device has access to the transmission medium.

medium access control (MAC) protocol. In a local area network, the protocol that governs access to the transmission medium, taking into account the topological aspects of the network, in order to enable the exchange of data between data stations. (T)

medium access control (MAC) sublayer. In a local area network, the part of the data link layer that applies a medium access method. The MAC sublayer supports topology-dependent functions and uses the services of the physical layer to provide services to the logical link control sublayer. (T)

metric. In Internet communications, a value, associated with a route, which is used to discriminate between multiple exit or entry points to the same autonomous system. The route with the lowest metric is preferred.

metropolitan area network (MAN). A network formed by the interconnection of two or more networks which may operate at higher speed than those networks, may cross administrative boundaries, and may use multiple access methods. (T) Contrast with *local area network (LAN)* and *wide area network (WAN)*.

MIB. (1) MIB module. (2) Management Information Base.

MIB object. Synonym for *MIB variable*.

MIB variable. In the Simple Network Management Protocol (SNMP), a specific instance of data defined in a MIB module. Synonymous with *MIB object*.

MIB view. In the Simple Network Management Protocol (SNMP), the collection of managed objects, known to the agent, that is visible to a particular community.

MILNET. The military network that was originally part of ARPANET. It was partitioned from ARPANET in 1984. MILNET provides a reliable network service for military installations.

modem (modulator/demodulator). (1) A functional unit that modulates and demodulates signals. One of the functions of a modem is to enable digital data to be transmitted over analog transmission facilities. (T) (A) (2) A device that converts digital data from a computer to an analog signal that can be transmitted on a telecommunication line, and converts the analog signal received to data for the computer.

modulo. (1) Pertaining to a modulus; for example, 9 is equivalent to 4 modulo 5. (2) See also *modulus*.

modulus. A number, such as a positive integer, in a relationship that divides the difference between two related numbers without leaving a remainder; for example, 9 and 4 have a modulus of 5 (9 - 4 = 5; 4 - 9 = -5; and 5 divides both 5 and -5 without leaving a remainder).

monitor. (1) A device that observes and records selected activities within a data processing system for analysis. Possible uses are to indicate significant departure from the norm, or to determine levels of utilization of particular functional units. (T) (2) Software or hardware that observes, supervises, controls, or verifies operations of a system. (A) (3) The function required to initiate the transmission of a token on the ring and to provide soft-error recovery in case of lost tokens, circulating frames, or other difficulties. The capability is present in all ring stations.

multicast. (1) Transmission of the same data to a selected group of destinations. (T) (2) A special form of broadcast in which copies of a packet are delivered to only a subset of all possible destinations.

multiple-domain support (MDS). A technique for transporting management services data between management services function sets over LU-LU and CP-CP sessions. See also *multiple-domain support message unit (MDS-MU)*.

multiple-domain support message unit (MDS-MU). The message unit that contains management services data and flows between management services function sets over the LU-LU and CP-CP sessions used by multiple-domain support. This message unit, as well as the actual management services data that it contains, is in general data stream (GDS) format. See also *control point management services unit (CP-MSU)*, *management services unit (MSU)*, and *network management vector transport (NMVT)*.

N

Name Binding Protocol (NBP). In AppleTalk networks, a protocol that provides name translation function from the AppleTalk entity (resource) name (character string) into an AppleTalk IP address (16-bit number) on the transport layer.

name resolution. In Internet communications, the process of mapping a machine name to the corresponding Internet Protocol (IP) address. See also *Domain Name System (DNS)*.

name server. In the Internet suite of protocols, synonym for *domain name server*.

nearest active upstream neighbor (NAUN). In the IBM Token-Ring Network, the station sending data directly to a given station on the ring.

neighbor. A router on a common subnetwork that has been designated by a network administrator to receive routing information.

NetBIOS. Network Basic Input/Output System. A standard interface to networks, IBM personal computers (PCs), and compatible PCs, that is used on LANs to provide message, print-server, and file-server functions. Application programs that use NetBIOS do not need to handle the details of LAN data link control (DLC) protocols.

network. (1) A configuration of data processing devices and software connected for information interchange. (2) A group of nodes and the links interconnecting them.

network accessible unit (NAU). A logical unit (LU), physical unit (PU), control point (CP), or system services control point (SSCP). It is the origin or the destination of information transmitted by the path control network. Synonymous with *network addressable unit*.

network address. According to ISO 7498-3, a name, unambiguous within the OSI environment, that identifies a set of network service access points.

network addressable unit (NAU). Synonym for *network accessible unit*.

network architecture. The logical structure and operating principles of a computer network. (T)

Note: The operating principles of a network include those of services, functions, and protocols.

network congestion. An undesirable overload condition caused by traffic in excess of what a network can handle.

network identifier. (1) In TCP/IP, that part of the IP address that defines a network. The length of the network ID depends on the type of network class (A, B, or C). (2) A 1- to 8-byte customer-selected name or an 8-byte IBM-registered name that uniquely identifies a specific subnetwork.

Network Information Center (NIC). In Internet communications, local, regional, and national groups throughout the world who provide assistance, documentation, training, and other services to users.

network layer. In Open Systems Interconnection (OSI) architecture, the layer that is responsible for routing, switching, and link-layer access across the OSI environment.

network management. The process of planning, organizing, and controlling a communication-oriented data processing or information system.

network management station. In the Simple Network Management Protocol (SNMP), a station that executes management application programs that monitor and control network elements.

network management vector transport (NMVT). A management services request/response unit (RU) that flows over an active session between physical unit management services and control point management services (SSCP-PU session).

network manager. A program or group of programs that is used to monitor, manage, and diagnose the problems of a network.

network node (NN). See *Advanced Peer-to-Peer Networking (APPN) network node*.

network user address (NUA). In X.25 communications, the X.121 address containing up to 15 binary code digits.

node. (1) In a network, a point at which one or more functional units connect channels or data circuits. (I) (2) Any device, attached to a network, that transmits and receives data.

noncanonical address. In LANs, a format for the transmission of medium access control (MAC) addresses for token-ring adapters. In noncanonical format, the most significant (leftmost) bit of each address byte is transmitted first. Contrast with *canonical address*.

Non-Return-to-Zero Changes-on-Ones Recording (NRZ-1). A recording method in which the ones are represented by a change in the condition of magnetization, and zeros are represented by the absence of change. Only the one signals are explicitly recorded. (Previously called ***non-return-to-zero inverted***, NRZI, recording.)

nonseed router. In AppleTalk networks, a router that acquires network number range and zone list information from a seed router attached to the same network.

O

Open Shortest Path First (OSPF). In the Internet suite of protocols, a function that provides intradomain information transfer. An alternative to the Routing Information Protocol (RIP), OSPF allows the lowest-cost routing and handles routing in large regional or corporate networks.

Open Systems Interconnection (OSI). (1) The interconnection of open systems in accordance with standards of the International Organization for Standardization (ISO) for the exchange of information. (T) (A) (2) The use of standardized procedures to enable the interconnection of data processing systems.

Note: OSI architecture establishes a framework for coordinating the development of current and future standards for the interconnection of computer systems. Network functions are divided into seven layers. Each layer represents a group of related data processing and communication functions that can be carried out in a standard way to support different applications.

Open Systems Interconnection (OSI) architecture. Network architecture that adheres to that particular set of ISO standards that relates to Open Systems Interconnection. (T)

Open Systems Interconnection (OSI) reference model. A model that describes the general principles of the Open Systems Interconnection, as well as the

purpose and the hierarchical arrangement of its seven layers. (T)

origin. An external logical unit (LU) or application program from which a message or other data originates. See also *destination*.

orphan circuit. A non-configured circuit whose availability is learned dynamically.

P

pacing. (1) A technique by which a receiving component controls the rate of transmission of a sending component to prevent overrun or congestion. (2) See also *flow control*, *receive pacing*, *send pacing*, *session-level pacing*, and *virtual route (VR) pacing*.

packet. In data communication, a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, control signals, and, possibly, error control information are arranged in a specific format. (I)

packet internet groper (PING). (1) In Internet communications, a program used in TCP/IP networks to test the ability to reach destinations by sending the destinations an Internet Control Message Protocol (ICMP) echo request and waiting for a reply. (2) In communications, a test of reachability.

packet loss ratio. The probability that a packet will not reach its destination or not reach it within a specified time.

packet mode operation. Synonym for *packet switching*.

packet switching. (1) The process of routing and transferring data by means of addressed packets so that a channel is occupied only during transmission of a packet. On completion of the transmission, the channel is made available for transfer of other packets. (I) (2) Synonymous with *packet mode operation*. See also *circuit switching*.

parallel bridges. A pair of bridges connected to the same LAN segment, creating redundant paths to the segment.

parallel transmission groups. Multiple transmission groups between adjacent nodes, with each group having a distinct transmission group number.

path. (1) In a network, any route between any two nodes. A path may include more than one branch. (T) (2) The series of transport network components (path control and data link control) that are traversed by the information exchanged between two network accessible units. See also *explicit route (ER)*, *route extension*, and *virtual route (VR)*.

path control (PC). The function that routes message units between network accessible units in the network and provides the paths between them. It converts the basic information units (BIUs) from transmission control (possibly segmenting them) into path information units (PIUs) and exchanges basic transmission units containing one or more PIUs with data link control. Path control differs by node type: some nodes (APPN nodes, for example) use locally generated session identifiers for routing, and others (subarea nodes) use network addresses for routing.

path cost. In link-state routing protocols, the sum of the link costs along the path between two nodes or networks.

path information unit (PIU). A message unit consisting of a transmission header (TH) alone, or a TH followed by a basic information unit (BIU) or a BIU segment.

pattern-matching character. A special character such as an asterisk (*) or a question mark (?) that can be used to represent one or more characters. Any character or set of characters can replace a pattern-matching character. Synonymous with *global character* and *wildcard character*.

permanent virtual circuit (PVC). In X.25 and frame-relay communications, a virtual circuit that has a logical channel permanently assigned to it at each data terminal equipment (DTE). Call-establishment protocols are not required. Contrast with *switched virtual circuit (SVC)*.

physical circuit. A circuit established without multiplexing. See also *data circuit*. Contrast with *virtual circuit*.

physical layer. In the Open Systems Interconnection reference model, the layer that provides the mechanical, electrical, functional, and procedural means to establish, maintain, and release physical connections over the transmission medium. (T)

physical unit (PU). (1) The component that manages and monitors the resources (such as attached links and adjacent link stations) associated with a node, as requested by an SSCP via an SSCP-PU session. An SSCP activates a session with the physical unit in order to indirectly manage, through the PU, resources of the node such as attached links. This term applies to type 2.0, type 4, and type 5 nodes only. (2) See also *peripheral PU* and *subarea PU*.

ping command. The command that sends an Internet Control Message Protocol (ICMP) echo-request packet

to a gateway, router, or host with the expectation of receiving a reply.

Point-to-Point Protocol (PPP). A protocol that provides a method for encapsulating and transmitting packets over serial point-to-point links.

polling. (1) On a multipoint connection or a point-to-point connection, the process whereby data stations are invited, one at a time, to transmit. (I) (2) Interrogation of devices for such purposes as to avoid contention, to determine operational status, or to determine readiness to send or receive data. (A)

port. (1) An access point for data entry or exit. (2) A connector on a device to which cables for other devices such as display stations and printers are attached. (3) The representation of a physical connection to the link hardware. A port is sometimes referred to as an adapter; however, there can be more than one port on an adapter. There may be one or more ports controlled by a single DLC process. (4) In the Internet suite of protocols, a 16-bit number used to communicate between TCP or the User Datagram Protocol (UDP) and a higher-level protocol or application. Some protocols, such as File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP), use the same well-known port number in all TCP/IP implementations. (5) An abstraction used by transport protocols to distinguish among multiple destinations within a host machine. (6) Synonymous with *socket*.

port number. In Internet communications, the identification of an application entity to the transport service.

private branch exchange (PBX). A private telephone exchange for transmission of calls to and from the public telephone network.

problem determination. The process of determining the source of a problem; for example, a program component, machine failure, telecommunication facilities, user or contractor-installed programs or equipment, environmental failure such as a power loss, or user error.

program temporary fix (PTF). A temporary solution or bypass of a problem diagnosed by IBM in a current unaltered release of the program.

protocol. (1) A set of semantic and syntactic rules that determine the behavior of functional units in achieving communication. (I) (2) In Open Systems Interconnection architecture, a set of semantic and syntactic rules that determine the behavior of entities in the same layer in performing communication functions. (T) (3) In SNA, the meanings of, and the sequencing rules for, requests and responses used for managing the network, transferring data, and synchronizing the states of network components. Synonymous with *line control discipline* and *line discipline*. See *bracket protocol* and *link protocol*.

protocol data unit (PDU). A unit of data specified in a protocol of a given layer and consisting of protocol control information of this layer, and possibly user data of this layer. (T)

R

Rapid Transport Protocol (RTP) connection. In high-performance routing (HPR), the connection established between the endpoints of the route to transport session traffic.

reachability. The ability of a node or a resource to communicate with another node or resource.

read-only memory (ROM). Memory in which stored data cannot be modified by the user except under special conditions.

real-time processing. The manipulation of data that are required, or generated, by some process while the process is in operation. Usually the results are used to influence the process, and perhaps related processes, while it is occurring.

reassembly. In communications, the process of putting segmented packets back together after they have been received.

receive not ready (RNR). In communications, a data link command or response that indicates a temporary condition of being unable to accept incoming frames.

receive not ready (RNR) packet. See *RNR packet*.

received line signal detector (RLSD). In the EIA 232 standard, a signal that indicates to the data terminal equipment (DTE) that it is receiving a signal from the remote data circuit-terminating equipment (DCE). Synonymous with *carrier detect* and *data carrier detect (DCD)*.

Recognized Private Operating Agency (RPOA). Any individual, company, or corporation, other than a government department or service, that operates a telecommunication service and is subject to the obligations undertaken in the Convention of the International Telecommunication Union and in the Regulations; for example, a communication common carrier.

reduced instruction-set computer (RISC). A computer that uses a small, simplified set of frequently used instructions for rapid execution.

remote. (1) Pertaining to a system, program, or device that is accessed through a telecommunication line. (2) Synonym for *link-attached*. (3) Contrast with *local*.

remote bridging. The function of a bridge that allows two bridges to connect multiple LANs using a telecommunication link. Contrast with *local bridging*.

Remote Execution Protocol (REXEC). A protocol that allows the execution of a command or program on any host in the network. The local host receives the results of the command execution.

Request for Comments (RFC). In Internet communications, the document series that describes a part of the Internet suite of protocols and related experiments. All Internet standards are documented as RFCs.

reset. On a virtual circuit, reinitialization of data flow control. At reset, all data in transit are eliminated.

reset request packet. In X.25 communications, a packet transmitted by the data terminal equipment (DTE) to the data circuit-terminating equipment (DCE) to request that a virtual call or a permanent virtual circuit be reset. The reason for the request can also be specified in the packet.

ring. See *ring network*.

ring network. (1) A network in which every node has exactly two branches connected to it and in which there are exactly two paths between any two nodes. (T) (2) A network configuration in which devices are connected by unidirectional transmission links to form a closed path.

ring segment. A section of a ring that can be isolated (by unplugging connectors) from the rest of the ring. See *LAN segment*.

rlogin (remote login). A service, offered by Berkeley UNIX-based systems, that allows authorized users of one machine to connect to other UNIX systems across an internet and interact as if their terminals were connected directly. The rlogin software passes information about the user's environment (for example, terminal type) to the remote machine.

RNR packet. A packet used by a data terminal equipment (DTE) or by a data circuit-terminating equipment (DCE) to indicate a temporary inability to accept additional packets for a virtual call or permanent virtual circuit.

root bridge. The bridge that is the root of a spanning tree formed between other active bridges in the bridging network. The root bridge originates and transmits bridge protocol data units (BPDUs) to other active bridges to maintain the spanning tree topology. It is the bridge with the highest priority in the network.

route. (1) An ordered sequence of nodes and transmission groups (TGs) that represent a path from an origin node to a destination node traversed by the traffic exchanged between them. (2) The path that network traffic uses to get from source to destination.

route bridge. A function of an IBM bridge program that allows two bridge computers to use a telecommunication link to connect two LANs. Each bridge computer is connected directly to one of the LANs, and the telecommunication link connects the two bridge computers.

route extension (REX). In SNA, the path control network components, including a peripheral link, that make up the portion of a path between a subarea node and a network addressable unit (NAU) in an adjacent peripheral node. See also *explicit route (ER)*, *path*, and *virtual route (VR)*.

Route Selection control vector (RSCV). A control vector that describes a route within an APPN network. The RSCV consists of an ordered sequence of control vectors that identify the TGs and nodes that make up the path from an origin node to a destination node.

router. (1) A computer that determines the path of network traffic flow. The path selection is made from several paths based on information obtained from specific protocols, algorithms that attempt to identify the shortest or best path, and other criteria such as metrics or protocol-specific destination addresses. (2) An attaching device that connects two LAN segments, which use similar or different architectures, at the reference model network layer. (3) In OSI terminology, a function that determines a path by which an entity can be reached. (4) In TCP/IP, synonymous with *gateway*. (5) Contrast with *bridge*.

routing. (1) The assignment of the path by which a message is to reach its destination. (2) In SNA, the forwarding of a message unit along a particular path through a network, as determined by parameters carried in the message unit, such as the destination network address in a transmission header.

routing domain. In Internet communications, a group of intermediate systems that use a routing protocol so that the representation of the overall network is the same within each intermediate system. Routing domains are connected to each other by exterior links.

Routing Information Protocol (RIP). In the Internet suite of protocols, an interior gateway protocol used to exchange intradomain routing information and to determine optimum routes between internet hosts. RIP

determines optimum routes on the basis of route metrics, not link transmission speed.

routing loop. A situation that occurs when routers circulate information among themselves until convergence occurs or until the networks involved are considered unreachable.

routing protocol. A technique used by a router to find other routers and to remain up to date about the best way to get to reachable networks.

routing table. A collection of routes used to direct datagram forwarding or to establish a connection. The information is passed among routers to identify network topology and destination feasibility.

Routing Table Maintenance Protocol (RTMP). In AppleTalk networks, a protocol that provides routing information generation and maintenance on the transport layer by means of the AppleTalk routing table. The AppleTalk routing table directs packet transmission through the internet from source socket to destination socket.

RouTing update Protocol (RTP). The VIrtual NEtworking System (VINES) protocol that maintains the routing database and allows the exchange of routing information between VINES nodes. See also *Internet Control Protocol (ICP)*.

rsh. A variant of the rlogin command that invokes a command interpreter on a remote UNIX machine and passes the command-line arguments to the command interpreter, skipping the login step completely.

S

SAP. See service access point.

seed router. In AppleTalk networks, a router that maintains configuration data (network range numbers and zone lists, for example) for the network. Each network must have at least one seed router. The seed router must be initially set up using the configurator tool. Contrast with *nonseed router*.

segment. (1) A section of cable between components or devices. A segment may consist of a single patch cable, several patch cables that are connected, or a combination of building cable and patch cables that are connected. (2) In Internet communications, the unit of transfer between TCP functions in different machines. Each segment contains control and data fields; the current byte-stream position and actual data bytes are identified along with a checksum to validate received data.

segmenting. In OSI, a function performed by a layer to map one protocol data unit (PDU) from the layer it supports into multiple PDUs.

sequence number. In communications, a number assigned to a particular frame or packet to control the transmission flow and receipt of data.

Serial Line Internet Protocol (SLIP). A protocol used over a point-to-point connection between two IP hosts over a serial line, for example, a serial cable or an RS232 connection into a modem, over a telephone line.

server. A functional unit that provides shared services to workstations over a network; for example, a file server, a print server, a mail server. (T)

service access point (SAP). (1) In Open Systems Interconnection (OSI) architecture, the point at which the services of a layer are provided by an entity of that layer to an entity of the next higher layer. (T) (2) A logical point made available by an adapter where information can be received and transmitted. A single service access point can have many links terminating in it.

Service Advertising Protocol (SAP). In Internetwork Packet Exchange (IPX), a protocol that provides the following:

- A mechanism that allows IPX servers on an internet to advertise their services by name and type. Servers using this protocol have their name, service type, and address recorded in all file servers running NetWare.
- A mechanism that allows a workstation to broadcast a query to discover the identities of all servers of all types, all servers of a specific type, or the nearest server of a specific type.
- A mechanism that allows a workstation to query any file server running NetWare to discover the names and addresses of all servers of a specific type.

session. (1) In network architecture, for the purpose of data communication between functional units, all the activities which take place during the establishment, maintenance, and release of the connection. (T) (2) A logical connection between two network accessible units (NAUs) that can be activated, tailored to provide various protocols, and deactivated, as requested. Each session is uniquely identified in a transmission header (TH) accompanying any transmissions exchanged during the session.

Simple Network Management Protocol (SNMP). In the Internet suite of protocols, a network management protocol that is used to monitor routers and attached networks. SNMP is an application layer protocol. Information on devices managed is defined and stored

in the application's Management Information Base (MIB).

SNA management services (SNA/MS). The services provided to assist in management of SNA networks.

socket. (1) An endpoint for communication between processes or application programs. (2) The abstraction provided by the University of California's Berkeley Software Distribution (commonly called Berkeley UNIX or BSD UNIX) that serves as an endpoint for communication between processes or applications.

source route bridging. In LANs, a bridging method that uses the routing information field in the IEEE 802.5 medium access control (MAC) header of a frame to determine which rings or token-ring segments the frame must transit. The routing information field is inserted into the MAC header by the source node. The information in the routing information field is derived from explorer packets generated by the source host.

source routing. In LANs, a method by which the sending station determines the route the frame will follow and includes the routing information with the frame. Bridges then read the routing information to determine whether they should forward the frame.

source service access point (SSAP). In SNA and TCP/IP, a logical address that allows a system to send data to a remote device from the appropriate communications support. Contrast with *destination service access point (DSAP)*.

spanning tree. In LAN contexts, the method by which bridges automatically develop a routing table and update that table in response to changing topology to ensure that there is only one route between any two LANs in the bridged network. This method prevents packet looping, where a packet returns in a circuitous route back to the sending router.

sphere of control (SOC). The set of control point domains served by a single management services focal point.

sphere of control (SOC) node. A node directly in the sphere of control of a focal point. A SOC node has exchanged management services capabilities with its focal point. An APPN end node can be a SOC node if it supports the function to exchange management services capabilities.

split horizon. A technique for minimizing the time to achieve network convergence. A router records the interface over which it received a particular route and does not propagate its information about the route back over the same interface.

spoofing. For data links, a technique in which a protocol initiated from an end station is acknowledged and processed by an intermediate node on behalf of the final destination. In IBM 6611 data link switching, for example, SNA frames are encapsulated into TCP/IP packets for transport across a non-SNA wide area network, unpacked by another IBM 6611, and passed to the final destination. A benefit of spoofing is the prevention of end-to-end session timeouts.

standard MIB. In the Simple Network Management Protocol (SNMP), a MIB module that is located under the management branch of the Structure of Management Information (SMI) and that is considered a standard by the Internet Engineering Task Force (IETF).

static route. The route between hosts, networks, or both that is manually entered into a routing table.

station. An input or output point of a system that uses telecommunication facilities; for example, one or more systems, computers, terminals, devices, and associated programs at a particular location that can send or receive data over a telecommunication line.

StreetTalk. In the VIrtual NEtworking System (VINES), a unique network-wide naming and addressing system that allows users to locate and access any resource on the network without knowing the network topology. See also *Internet Control Protocol (ICP)* and *RouTing update Protocol (RTP)*.

Structure of Management Information (SMI). (1) In the Simple Network Management Protocol (SNMP), the rules used to define the objects that can be accessed by means of a network management protocol. (2) In OSI, the set of standards relating to management information. The set includes the *Management Information Model* and the *Guidelines for the Definition of Managed Objects*

subarea. A portion of the SNA network consisting of a subarea node, attached peripheral nodes, and associated resources. Within a subarea node, all network accessible units (NAUs), links, and adjacent link stations (in attached peripheral or subarea nodes) that are addressable within the subarea share a common subarea address and have distinct element addresses.

subnet. (1) In TCP/IP, a part of a network that is identified by a portion of the IP address. (2) Synonym for *subnetwork*.

subnet address. In Internet communications, an extension to the basic IP addressing scheme where a portion of the host address is interpreted as the local network address.

subnet mask. Synonym for *address mask*.

subnetwork. (1) Any group of nodes that have a set of common characteristics, such as the same network ID. (2) Synonymous with *subnet*.

Subnetwork Access Protocol (SNAP). In LANs, a 5-byte protocol discriminator that identifies the non-IEEE standard protocol family to which a packet belongs. The SNAP value is used to differentiate between protocols that use $AA as their service access point (SAP) value.

subnetwork mask. Synonym for *address mask*.

subsystem. A secondary or subordinate system, usually capable of operating independently of, or asynchronously with, a controlling system. (T)

switched virtual circuit (SVC). An X.25 circuit that is dynamically established when needed. The X.25 equivalent of a switched line. Contrast with *permanent virtual circuit (PVC)*.

synchronous. (1) Pertaining to two or more processes that depend upon the occurrence of specific events such as common timing signals. (T) (2) Occurring with a regular or predictable time relationship.

Synchronous Data Link Control (SDLC). (1) A discipline conforming to subsets of the Advanced Data Communication Control Procedures (ADCCP) of the American National Standards Institute (ANSI) and High-level Data Link Control (HDLC) of the International Organization for Standardization, for managing synchronous, code-transparent, serial-by-bit information transfer over a link connection. Transmission exchanges may be duplex or half-duplex over switched or nonswitched links. The configuration of the link connection may be point-to-point, multipoint, or loop. (I) (2) Contrast with *binary synchronous communication (BSC)*.

SYNTAX. In the Simple Network Management Protocol (SNMP), a clause in the MIB module that defines the abstract data structure that corresponds to a managed object.

system. In data processing, a collection of people, machines, and methods organized to accomplish a set of specific functions. (I) (A)

system configuration. A process that specifies the devices and programs that form a particular data processing system.

system services control point (SSCP). A component within a subarea network for managing the configuration, coordinating network operator and problem determination requests, and providing directory services and other session services for users of the network. Multiple SSCPs, cooperating as peers with one another, can divide the network into domains of control, with each SSCP having a hierarchical control relationship to the physical units and logical units within its own domain.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks. The layered structure of SNA allows the ultimate origins and destinations of information, that is, the users, to be independent of and unaffected by the specific SNA network services and facilities used for information exchange.

T

TCP/IP. (1) Transmission Control Protocol/Internet Protocol. (2) A UNIX-like/Ethernet-based system-interconnect protocol originally developed by the US Department of Defense. TCP/IP facilitated ARPANET (Advanced Research Projects Agency Network), a packet-switched research network for which layer 4 was TCP and layer 3, IP.

Telnet. In the Internet suite of protocols, a protocol that provides remote terminal connection service. It allows users of one host to log on to a remote host and interact as directly attached terminal users of that host.

threshold. (1) In IBM bridge programs, a value set for the maximum number of frames that are not forwarded across a bridge due to errors, before a “threshold exceeded” occurrence is counted and indicated to network management programs. (2) An initial value from which a counter is decremented to 0, or a value to which a counter is incremented or decremented from an initial value.

throughput class. In packet switching, the speed at which data terminal equipment (DTE) packets travel through the packet switching network.

time to live (TTL). A technique used by best-effort delivery protocols to inhibit endlessly looping packets. The packet is discarded if the TTL counter reaches 0.

timeout. (1) An event that occurs at the end of a predetermined period of time that began at the occurrence of another specified event. (I) (2) A time interval allotted for certain operations to occur; for example, response to polling or addressing before system operation is interrupted and must be restarted.

token. (1) In a local area network, the symbol of authority passed successively from one data station to another to indicate the station temporarily in control of the transmission medium. Each data station has an opportunity to acquire and use the token to control the medium. A token is a particular message or bit pattern that signifies permission to transmit. (T) (2) In LANs,

a sequence of bits passed from one device to another along the transmission medium. When the token has data appended to it, it becomes a frame.

token ring. (1) According to IEEE 802.5, network technology that controls media access by passing a token (special packet or frame) between media-attached stations. (2) A FDDI or IEEE 802.5 network with a ring topology that passes tokens from one attaching ring station (node) to another. (3) See also *local area network (LAN)*.

token-ring network. (1) A ring network that allows unidirectional data transmission between data stations, by a token passing procedure, such that the transmitted data return to the transmitting station. (T) (2) A network that uses a ring topology, in which tokens are passed in a circuit from node to node. A node that is ready to send can capture the token and insert data for transmission.

topology. In communications, the physical or logical arrangement of nodes in a network, especially the relationships among nodes and the links between them.

topology database update (TDU). A message about a new or changed link or node that is broadcast among APPN network nodes to maintain the network topology database, which is fully replicated in each network node. A TDU contains information that identifies the following:

- The sending node
- The node and link characteristics of various resources in the network
- The sequence number of the most recent update for each of the resources described.

trace. (1) A record of the execution of a computer program. It exhibits the sequences in which the instructions were executed. (A) (2) For data links, a record of the frames and bytes transmitted or received.

transceiver (transmitter-receiver). In LANs, a physical device that connects a host interface to a local area network, such as Ethernet. Ethernet transceivers contain electronics that apply signals to the cable and that sense collisions.

Transmission Control Protocol (TCP). A communications protocol used in the Internet and in any network that follows the U.S. Department of Defense standards for internetwork protocol. TCP provides a reliable host-to-host protocol between hosts in packet-switched communications networks and in interconnected systems of such networks. It uses the Internet Protocol (IP) as the underlying protocol.

Transmission Control Protocol/Internet Protocol (TCP/IP). A set of communications protocols that support peer-to-peer connectivity functions for both local and wide area networks.

transmission group (TG). (1) A connection between adjacent nodes that is identified by a transmission group number. (2) In a subarea network, a single link or a group of links between adjacent nodes. When a transmission group consists of a group of links, the links are viewed as a single logical link, and the transmission group is called a *multilink transmission group (MLTG)*. A *mixed-media multilink transmission group (MMMLTG)* is one that contains links of different medium types (for example, token-ring, switched SDLC, nonswitched SDLC, and frame-relay links). (3) In an APPN network, a single link between adjacent nodes. (4) See also *parallel transmission groups*.

transmission header (TH). Control information, optionally followed by a basic information unit (BIU) or a BIU segment, that is created and used by path control to route message units and to control their flow within the network. See also *path information unit*.

transparent bridging. In LANs, a method for tying individual local area networks together through the medium access control (MAC) level. A transparent bridge stores the tables that contain MAC addresses so that frames seen by the bridge can be forwarded to another LAN if the tables indicate to do so.

transport layer. In the Open Systems Interconnection reference model, the layer that provides a reliable end-to-end data transfer service. There may be relay open systems in the path. (T) See also *Open Systems Interconnection reference model*.

trap. In the Simple Network Management Protocol (SNMP), a message sent by a managed node (agent function) to a management station to report an exception condition.

tunneling. To treat a transport network as though it were a single communication link or LAN. See also *encapsulation*.

T1. In the United States, a 1.544-Mbps public access line. It is available in twenty-four 64-Kbps channels. The European version (E1) transmits 2.048 Mbps.

U

universally administered address. In a local area network, the address permanently encoded in an adapter at the time of manufacture. All universally administered addresses are unique. Contrast with *locally administered address*.

User Datagram Protocol (UDP). In the Internet suite of protocols, a protocol that provides unreliable, connectionless datagram service. It enables an

application program on one machine or process to send a datagram to an application program on another machine or process. UDP uses the Internet Protocol (IP) to deliver datagrams.

V

V.24. In data communication, a specification of the CCITT that defines the list of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE).

V.25. In data communication, a specification of the CCITT that defines the automatic answering equipment and parallel automatic calling equipment on the General Switched Telephone Network, including procedures for disabling of echo controlled devices for both manually and automatically established calls.

V.35. In data communication, a specification of the CCITT that defines the list of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) at various data rates.

V.36. In data communication, a specification of the CCITT that defines the list of definitions for interchange circuits between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) at rates of 48, 56, 64, or 72 kilobits per second.

version. A separately licensed program that usually has significant new code or new function.

VINES. VIrtual NEtworking System.

virtual circuit. (1) In packet switching, the facilities provided by a network that give the appearance to the user of an actual connection. (T) See also *data circuit*. Contrast with *physical circuit*. (2) A logical connection established between two DTEs.

virtual connection. In frame relay, the return path of a potential connection.

virtual link. In Open Shortest Path First (OSPF), a point-to-point interface that connects border routers that are separated by a non-backbone transit area. Because area routers are part of the OSPF backbone, the virtual link connects the backbone. The virtual links ensure that the OSPF backbone does not become discontinuous.

VIrtual NEtworking System (VINES). The network operating system and network software from Banyan Systems, Inc. In a VINES network, virtual linking allows all devices and services to appear to be directly connected to each other, when they may actually be thousands of miles apart. See also *StreetTalk*.

virtual route (VR). (1) In SNA, either (a) a logical connection between two subarea nodes that is physically realized as a particular explicit route or (b) a logical connection that is contained wholly within a subarea node for intranode sessions. A virtual route between distinct subarea nodes imposes a transmission priority on the underlying explicit route, provides flow control through virtual route pacing, and provides data integrity through sequence numbering of path information units (PIUs). (2) Contrast with *explicit route (ER)*. See also *path* and *route extension (REX)*.

W

wide area network (WAN). (1) A network that provides communication services to a geographic area larger than that served by a local area network or a metropolitan area network, and that may use or provide public communication facilities. (T) (2) A data communication network designed to serve an area of hundreds or thousands of miles; for example, public and private packet-switching networks, and national telephone networks. (3) Contrast with *local area network (LAN)* and *metropolitan area network (MAN)*.

wildcard character. Synonym for *pattern-matching character*.

X

X.21. An International Telegraph and Telephone Consultative Committee (CCITT) recommendation for a general-purpose interface between data terminal equipment and data circuit-terminating equipment for synchronous operations on a public data network.

X.25. (1) An International Telegraph and Telephone Consultative Committee (CCITT) recommendation for the interface between data terminal equipment and packet-switched data networks. (2) See also *packet switching*.

Xerox Network Systems (XNS). The suite of internet protocols developed by the Xerox Corporation. Although similar to TCP/IP protocols, XNS uses different packet formats and terminology. See also *Internetwork Packet Exchange (IPX)*.

Z

zone. In AppleTalk networks, a subset of nodes within an internet.

Zone Information Protocol (ZIP). In AppleTalk networks, a protocol that provides zone management service by maintaining a mapping of the zone names and network numbers across the internet on the session layer.

zone information table (ZIT). A listing of network numbers and their associated zone name mappings in the internet. This listing is maintained by each internet router in an AppleTalk internet.

Index

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

X

Tell Us What You Think!

Nways Multiprotocol Routing Services
Software User's Guide
Version 2.1

Publication No. SC30-3681-05

We hope you find this publication useful, readable, and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form. If you are in the U.S.A., you can mail this form postage free or fax it to us at 1-800-253-3520. Elsewhere, your local IBM branch office or representative will forward your comments or you may mail them directly to us.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific comments or problems:

Please tell us how we can improve this book:

Thank you for your comments. If you would like a reply, provide the necessary information below.

Name

Address

Company or Organization

Phone No.

Tell Us What You Think!
SC30-3681-05

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Design & Information Development
Dept. CGF/Bldg. 656
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

SC30-3681-05

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC30-3681-05