8371 Networking Multilayer Ethernet Switch

Software User's Guide and Configuration Reference

GC30-9688-01

8371 Networking Multilayer Ethernet Switch

Software User's Guide and Configuration Reference

GC30-9688-01

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

Second Edition (July 1999)

This edition applies to Version 5.0 of the IBM 8371 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:

International Business Machines Corporation
Design and Information Development
Department CGF
P.O. Box 12195
Research Triangle Park, NC 27709-9990
U.S.A.

Alternatively, you can access the Web at this website to submit your comments online:
http://www.networking.ibm.com/feedback/pubsurv.html

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.

© Copyright International Business Machines Corporation 1996, 1999. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 1996, 1999

Figures

© Copyright IBM Corp. 1996, 1999

Tables

© Copyright IBM Corp. 1996, 1999

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, North Castle Drive, Armonk, NY 10504-1785, 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.

© Copyright IBM Corp. 1996, 1999

Notice to Users of Online Versions of This Book

For online versions of this book, you are authorized to:

- Copy, modify, and print the documentation contained on the media, for use within your enterprise, provided you reproduce the copyright notice, all warning statements, and other required statements on each copy or partial copy.
- Transfer the original unaltered copy of the documentation when you transfer the related IBM product (which may be either machines you own, or programs, if the program's license terms permit a transfer). You must, at the same time, destroy all other copies of the documentation.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization.

THERE ARE NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Your failure to comply with the terms above terminates this authorization. Upon termination, you must destroy your machine-readable documentation.

© Copyright IBM Corp. 1996, 1999

Trademarks

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

Advanced Peer-to-Peer Networking	CUA	Operating System/2
AIX	IBM	RS/6000
AIXwindows	Micro Channel	System/370
APPN	NetView	VTAM
BookManager	Nways	Web Explorer
Common User Access	OS/2	PS/2

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 logo are trademarks or registered trademarks of Microsoft Corporation.

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

© Copyright IBM Corp. 1996, 1999

Preface

This manual contains the information you will need to use the command line interface for configuration and operation of the IBM 8371, hereafter referred to as the switch. With the help of this manual, you should be able to perform the following processes and operations:

- Configure, monitor, and use the base code on your 8371
- Configure, monitor, and use the interfaces and Link Layer software supported by the switch.

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.

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 underlined as shown in the following example:

   reload

   In this example, you can enter either the whole command (reload) or its abbreviation (rel).
2. Keyword choices for a parameter are enclosed in brackets and separated by the word or. For example:

   ```
   command [keyword1 or keyword2]
   ```

   Choose one of the keywords as a value for the parameter.
3. 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.
4. 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.
5. Keyboard key combinations are indicated in text in the following way::

   Ctrl-P

   The key combination **Ctrl P** indicates that you should press the Ctrl key and the P key simultaneously. In certain circumstances, this key combination changes the command line prompt.
6. Names of keyboard keys are indicated like this: **Enter**

© Copyright IBM Corp. 1996, 1999

8371 Library

The following publication is shipped in displayable softcopy form on the 8371 CD-ROM (SK2T-0446-00). This CD-ROM is shipped with initial orders for the IBM 8371.

- *Networking Multilayer Ethernet Switch Installation and Planning Guide*, GA27-4226-00

Part 1. User's Guide

© Copyright IBM Corp. 1996, 1999

Chapter 1. Getting Started

This chapter shows you how to get started with using the following components related to the IBM Multilayer Ethernet Switch (IBM 8371):

- Device console terminals
- Device software (IBM 8371)
- Device software user interface

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 device is installed correctly.

Have you...

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

  Attention: If you are using a service port-attached terminal to configure or monitor your IBM 8371 and your service terminal is unreadable, you need to change some parameters in your configuration. (See "Service Terminal Display Unreadable" in 8371 Networking Multilayer Ethernet Switch Installation and Planning Guide.)

  Refer to your hardware documentation.
- Connected your device 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 *Networking Multilayer Ethernet Switch Installation and Planning Guide*.

Loading the 8371 and the 8371 Blade Operational Code

ATTENTION: To download the latest operational code for the 8371, or to see Tech Tips, customer forums, or to register for information updates, go to:

www.networking.ibm.com/support/products.nsf/techsupport/(8371)?OpenDocument

Loading the 8265 CP/Switch Code for 8371 Blade (8265 Version)

You need to download Control Point Switch code from the Web for 8265 Blade from

www.networking.ibm.com/support/products.nsf/techsupport/(8265)?OpenDocument

The microcode level should be **PNNI Version 4.1.2** or higher.

© Copyright IBM Corp. 1996, 1999

Loading the 8260 CP/Switch Code for 8371 Blade (8260 Version)

If you use a Control Point Switch code in your 8260, you need to download the Control Point Switch code from the Web for 8260 Blade from

www.networking.ibm.com/support/products.nsf/techsupport/(8260)?OpenDocument

The microcode level should be **Version 2.5.4** or higher.

Accessing the Software Using Local and Remote Consoles

The device console lets you use the device user interface to monitor and change the function of the device's networking software. The device 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 device. You may need to use a local console during the initial software installation. After the initial setup connection, you can connect using Telnet, as long as IP forwarding has been enabled. (Refer to *Protocols and Features* for more information on enabling IP forwarding.)

When the configured device 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 device is ready to accept OPCON commands.

Your IBM 8371 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 IBM 8371 was not pre-configured at the factory, you will need to use an ASCII terminal attached to the 8371 service port to initially configure it.

Important: Garbage, random characters, reverse question marks, or the inability to connect your terminal to the 8371 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 8371.

  If the IBM 8371 is set to a specific baud rate, the terminal or terminal emulator must be set to the same baud rate.

  If the IBM 8371 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).

  Refer to your hardware documentation for more information.
- Defective terminal or device (ac) grounds.
- Defective, incorrectly shielded, or incorrectly grounded EIA 232 (RS-232) cable between the terminal and the IBM 8371.
- Defective terminal or terminal emulator.
- Defective IBM 8371 system board.
- High ambient electromagnetic interference (EMI) levels.

- Power line disturbances.

(See “Service Terminal Display Unreadable” in the *8371 Networking Multilayer Ethernet Switch Installation and Planning Guide* .)

Once the IBM 8371 is initially configured, you will not need a local console for device operation, as long as IP is enabled.

The device 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 *Networking Multilayer Ethernet Switch Installation and Planning Guide*.

Remote Consoles

Remote consoles attach to the device 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 8371 was not pre-configured at the factory.

Telnet Connections

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

Remote Login Names and Passwords

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

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 device by starting Telnet on your host system. To log in remotely, begin at step 1. To log in locally, begin at step 3.

To log in from a remote console:

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

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

   ```
   % telnet brandenburg
   ```

   or

   ```
   % telnet 128.185.132.43
   ```

   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 device. 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 57. Remember to reload to activate any changes.

Note: If you do not enter a login name and valid password within 1 minute of the initial prompt, or if you enter an incorrect password three times in succession, the device drops the Telnet connection.

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

   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 reenter a command. See “Command Completion” on page 16 and “Command History” on page 19 for more information.

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

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

5. Exit the device as described in “Exiting the Device”.

Reloading the Device

Use the **reload** command to reboot the device by loading a new copy of the configuration from memory. Whenever you change a user-configurable parameter that is not dynamically configurable, you must reload the device for the change to take effect. For example:

```
* reload

The configuration has been changed, save it? (Yes or [No] or Abort)

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

Exiting the Device

Return to the * prompt and use the **logout** command to 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 software is a multitasking system that schedules use of the CPU among various processes and hardware devices. The device software:

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

Understanding the First-Level User Interface

The user interface to the software consists of the main menu (process) and several subsidiary menus (processes). These menus are related to the multiple levels of processes in the software.

The first level of processes consists of the OPCON and CONFIG-ONLY processes. In most cases, you will use the OPCON process to access the second level to configure or operate the base services, features, interfaces, and protocols you will run on your IBM 8371.

The second level contains processes such as Configuration (CONFIG), Console (GWCON) and Event Logging System (MONITR). You may use the OPCON commands **configuration**, **console** or **event** to access these second level processes. Alternatively, you may use the **status** command to list the second level processes and then use the **talk** *pid* command to access the second-level processes. There are processes that you cannot use in the software. See Table 1 on page 10 for an overview of the processes.

Figure 1 shows the processes and how they fit within the structure of the device software.

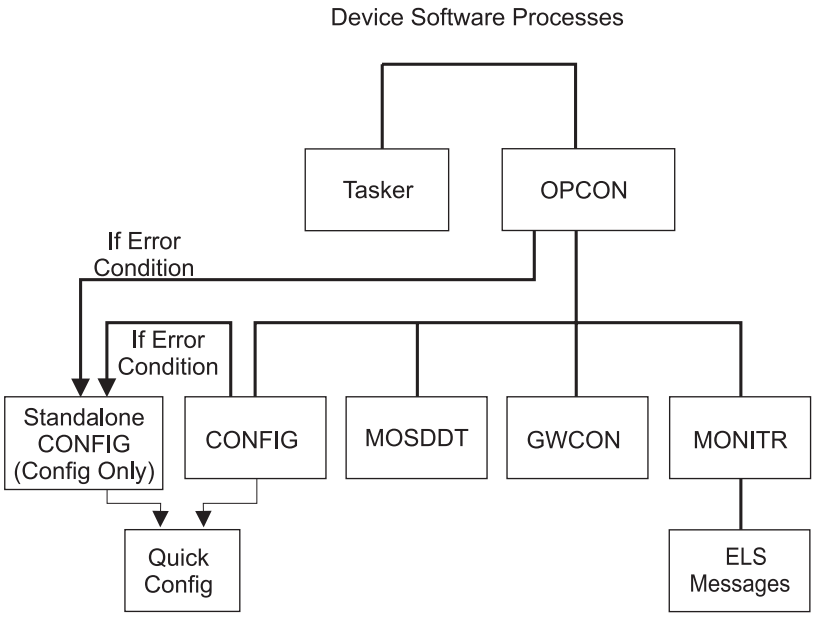

Figure 1. IBM 8371

Figure 2 on page 8 is an example of the relationship between the various process levels.

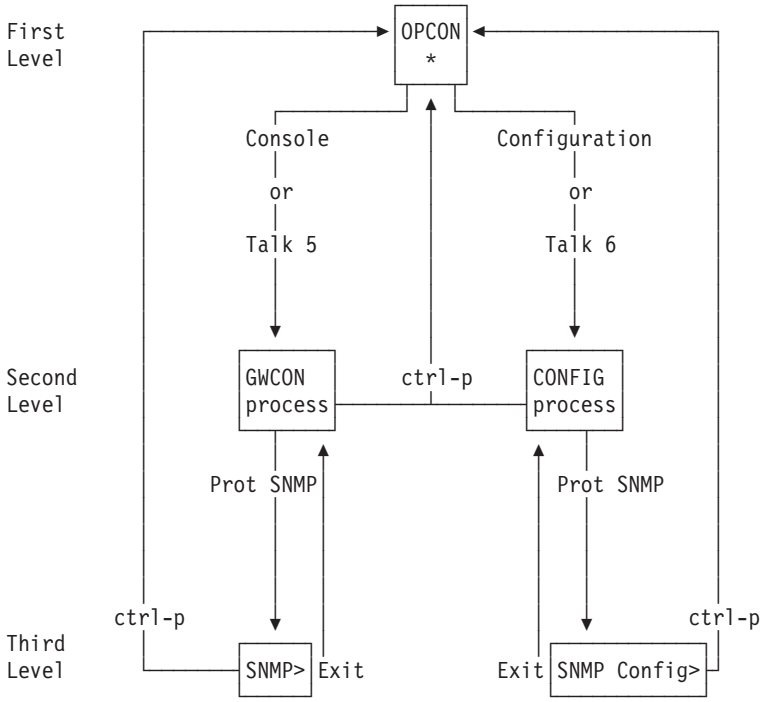

Figure 2. Relationship of Processes and Commands

Note: Also shown in Figure 2 are the various commands to access each process level and return from each process level.

See “What is the OPCON Process?” on page 41 for more information about OPCON.

The ROPCON process handles processing from remote consoles and is essentially the same as the OPCON process.

System Security

Multiple users with login permissions can be added using the **add user** command. See “Configuring User Access” on page 55 for details on security issues and for information on the **set password** and **add user** commands.

Chapter 2. Using the Software

This chapter describes how to use the software. It consists of:

Entering Commands

When typing a command, remember the following:

- You may type only enough sequential letters of the command to make it unique among the available commands. For example, to execute the **reload** command you must enter **rel** as a minimum. The minimum number of required characters are underlined in the command syntax chapters.
- 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 to be executed.
- When command completion is enabled, you can press Esc and enter **?** to obtain help on entering commands. See "Command Completion" on page 16 and "Command History" on page 19 for more information.

Connecting to a Process

When you start the device, 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.

Commands that are needed more often appear before the "- - - - -" separator. Enter the appropriate command at the OPCON prompt (*). See Table 11 on page 42 for a list of commands.

Alternatively, you can:

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 device processes, such as the process IDs (PIDs), process names and status of the process. Issuing the **status** command is shown in the following example:
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 "What is the OPCON Process?" on page 41.)

   Note: Not every process listed has a user interface (for example, the **talk 3** process). The **talk 4** command is for use by IBM service representatives.

© Copyright IBM Corp. 1996, 1999

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 **Enter** again.)

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

Table 1. Processes, Their Purpose, and Commands to Access

Process	Level and Purpose	Command to Access	Input Prompt
OPCON	Level 1 - access to all secondary levels	**Ctrl-P**	asterisk (*)
CONFIG	Level 2 - base services configuration and access to configuration third level	**Configuration or talk 6**	Config >
GWCON	Level 2 - base services operation and monitoring and access to operations and monitoring on third level	**Console or talk 5**	plus sign (+)
MONITR	Level 2 - message display	**Event or talk 2**	(none)
ELSCon	Level 2 - direct monitoring and access to ELS console	**els or talk 7**	ELS Secondary Console>
MOSDBG	Level 2 - diagnostic environment	**talk 4**	db>
Note: Only enter the **talk 4** command under the direction of a service representative.			

At the OPCON 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 reenter a command. See “Command Completion” on page 16 and “Command History” on page 19 for additional details or press **Escape ?**.

Getting Help

At the command prompts, you can obtain help in the form of a listing of the commands available at that level. To do this, type **?** (the **help** command), and then press **Enter**. Use **?** to list the commands that are available from the current level. You can usually enter a **?** after a specific command name to list its options.

For example, the following information appears if you enter **?** at the * prompt:

```
*?
        CONFIGURATION         (Talk 6)
        CONSOLE                 (Talk 5)
        EVENT Logging System  (Talk 2)
        ELS Console              (Talk 7)
        LOGOUT
        PING (IP-Address)
        RELOAD
        RESTART
        TELNET to IP-Address (this terminal type)
-----------------------------------------------
        DIVERT output from process
        FLUSH output from process
        HALT output from process
        INTERCEPT character is
        MEMORY statistics
```

```
        STATUS of Processes(es)
        TALK to process
(you may cycle through these commands by pressing the TAB key)
```

Exiting a Lower Level Environment

The multiple-level nature of the software places you in secondary, tertiary, and even lower level environments as you configure or operate the 8371. To return to the next higher level, enter the **exit** command. To get to the secondary level, continue entering **exit** until you receive the secondary level prompt (either `Config>` or `+`).

For example, to exit the ASRT protocol configuration process:

```
ASRT config> exit
Config>
```

If you need to get to the primary level (OPCON), enter the intercept character (**Ctrl-P** by default).

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 console (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 the default *intercept character*.

If you use the intercept character from a third-level or lower level menu to return to the * prompt, the next time you use the **talk** command to talk to the same process, you will reenter that same level menu. This link goes away when the device is re-initialized.

Accessing the Second-Level Processes

All interfaces, features, and protocols have commands that you use to access the following processes:

- The configuration process to initially configure and enable the interface, feature, or protocol, as well as perform later configuration changes.
- The operating/monitoring process to display information about each interface, feature, or protocol, to make temporary configuration changes, or to activate configuration changes.

You can also configure or operate some base system services through the second-level processes. The commands to perform these functions are described starting in “What is CONFIG?” on page 53.

The next sections describe the procedures for accessing the second-level processes.

Accessing the Configuration Process, CONFIG (Talk 6)

Each protocol configuration process is accessed through the device’s CONFIG process. CONFIG is the second-level process of the device 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 of menus. Protocol configuration command interfaces are menus within the CONFIG interface. Each protocol configuration interface has its own prompt. For example, the prompt for the SNMP protocol command interface is `SNMP config>`.

The next sections describe these procedures in more detail.

Entering the CONFIG Process

To enter the CONFIG process from OPCON and obtain the CONFIG prompt, enter the **configuration** command. Alternatively, you can enter the OPCON **talk** command and the PID for CONFIG. The PID for CONFIG is 6.

```
* configuration
```

or

```
* talk 6
```

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

Reloading the Device

Changes that you make to the protocol parameters through CONFIG do not take effect until you either activate the net that contains any dynamic changes or reload the device software.

To reload the device, enter the OPCON **reload** command. For example:

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

Accessing the Console Operating/Monitoring Process, GWCON (Talk 5)

To view information about the interfaces, features, or protocols or to change parameters while running, you must access and use the operating (monitoring) process. Operating command interfaces are modes of the GWCON interface. Within the GWCON mode, each interface, feature, or protocol interface has its own prompt. For example, the prompt for the SNMP protocol is `SNMP>`.

Note: Any parameters you change in this process will not remain active across any event that causes the 8371 to reload the operational code, such as a power outage or entering the **reload** command.

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, enter the **console** command. Alternatively, you may enter the **talk** command and the PID for GWCON. The PID for GWCON is 5. For example:

```
* console
```

or

```
* talk 5
```

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

Accessing the Secondary ELS Console Process, ELSCon (Talk 7)

The Secondary ELS Console provides convenient access to GWCON `talk 5` ELS without disrupting the current state of GWCON. You may be in the middle of a **ping** in `talk 5`, or deep inside a `talk 5` menu structure, and want to control ELS without disrupting the current state of GWCON. The secondary ELS console (Talk 7) serves this purpose.

To enter the Secondary ELS Console (ELScon) process from OPCON and obtain the Secondary ELS Console prompt, enter the **els** command. Alternatively, you may enter the **talk 7** command.

In the following example, another ELS event is displayed while performing a **ping** command.

Note: The intercept character (Ctrl-P by default) is used to obtain the OPCON prompt (*).

```
*talk 5
+protocol hst
HST>ping 10.0.0.9
PING 10.0.0.2 -> 10.0.0.9: 56 data bytes, ttl=64, every 1 sec.

*talk 7

ELS Secondary Console>display event ip.7
Complete
ELS Secondary Console>
*talk 2
00:20:48   IP.007: 10.0.0.2 -> 10.0.0.9
00:20:49   IP.007: 10.0.0.2 -> 10.0.0.9
```

Accessing the Third-Level Processes

After accessing the second level, you must enter commands on the third level to configure or operate the interfaces, features, and protocols in your IBM 8371. The following sections describe how to access the third level processes.

Accessing Feature Configuration and Operating Processes

To help you access the IBM 8371 feature configuration and operating processes, this section outlines both of these procedures.

Accessing the Feature Processes

Use the **feature** command from the CONFIG process to access configuration commands for specific IBM 8371 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 ?

QOS
```

```
Self Learning IP
RMON
Feature name or number [Self Learning IP] ?
```

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

```
Config> feature self learning ip

Self Learning IP configuration

Self Learning IP Config>
```

Table 15 on page 63 lists the available feature numbers and names.

Once you access the configuration or operating 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 Protocol Configuration and Operating Processes

This section describes how to access the protocol configuration and operating processes.

Entering a 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 page 63 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, SNMP) 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 SNMP protocol configuration process:

   ```
   Config> protocol SNMP
   ```

   or

   ```
   Config> protocol 11
   SNMP user configuration
   ```

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

   ```
   SNMP config>
   ```

You can now begin entering the protocol's configuration commands. See the corresponding protocol section of the *Protocols and Features* 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 device. 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.

Entering a Protocol Operating 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 device. For example:

```
+ configuration

Num Name  Protocol
11  SNMP  Simple Network Management Protocol
23  ASRT  Adaptive Source Routing Transparent Enhanced Bridge
29  MPOA  Multi-Protocol Over ATM
Num Name  Feature
6   QOS   Quality of Service
17  Self  Self Learning IP
18  RMON  Remote Network Monitor

64 Total Networks:
Net Interface  MAC/Data-Link         Hardware         State
0   Eth/0      Ethernet/IEEE 802.3   10/100 Ethernet  Up
1   Eth/1      Ethernet/IEEE 802.3   10/100 Ethernet  Up
2   Eth/2      Ethernet/IEEE 802.3   10/100 Ethernet  Up
3   Eth/3      Ethernet/IEEE 802.3   10/100 Ethernet  Up
4   Eth/4      Ethernet/IEEE 802.3   10/100 Ethernet  Up
5   Eth/5      Ethernet/IEEE 802.3   10/100 Ethernet  Up
6   Eth/6      Ethernet/IEEE 802.3   10/100 Ethernet  Up
7   Eth/7      Ethernet/IEEE 802.3   10/100 Ethernet  Up
8   Eth/8      Ethernet/IEEE 802.3   10/100 Ethernet  Up
9   Eth/9      Ethernet/IEEE 802.3   10/100 Ethernet  Up
10  Eth/10     Ethernet/IEEE 802.3   10/100 Ethernet  Up
11  Eth/11     Ethernet/IEEE 802.3   10/100 Ethernet  Up
12  Eth/12     Ethernet/IEEE 802.3   10/100 Ethernet  Up
13  Eth/13     Ethernet/IEEE 802.3   10/100 Ethernet  Up
14  Eth/14     Ethernet/IEEE 802.3   10/100 Ethernet  Up
15  Eth/15     Ethernet/IEEE 802.3   10/100 Ethernet  Up
16  Eth/16     Ethernet/IEEE 802.3   10/100 Ethernet  Up
17  Eth/17     Ethernet/IEEE 802.3   10/100 Ethernet  Up
17  Eth/18     Ethernet/IEEE 802.3   10/100 Ethernet  Up
19  Eth/19     Ethernet/IEEE 802.3   10/100 Ethernet  Up
20  Eth/20     Ethernet/IEEE 802.3   10/100 Ethernet  Up
21  Eth/21     Ethernet/IEEE 802.3   10/100 Ethernet  Up
22  Eth/22     Ethernet/IEEE 802.3   10/100 Ethernet  Up
23  Eth/23     Ethernet/IEEE 802.3   10/100 Ethernet  Up
24  Eth/24     Ethernet/IEEE 802.3   10/100 Ethernet  Up
25  Eth/25     Ethernet/IEEE 802.3   10/100 Ethernet  Up
26  Eth/26     Ethernet/IEEE 802.3   10/100 Ethernet  Up
27  Eth/27     Ethernet/IEEE 802.3   10/100 Ethernet  Up
28  Eth/28     Ethernet/IEEE 802.3   10/100 Ethernet  Up
29  Eth/29     Ethernet/IEEE 802.3   10/100 Ethernet  Up
30  Eth/30     Ethernet/IEEE 802.3   10/100 Ethernet  Up
31  Eth/31     Ethernet/IEEE 802.3   10/100 Ethernet  Up
32  Eth/31     Ethernet/IEEE 802.3   10/100 Ethernet  Up
33  Eth/31     Ethernet/IEEE 802.3   10/100 Ethernet  Up
34  Eth/31     Ethernet/IEEE 802.3   10/100 Ethernet  Up
35  Eth/31     Ethernet/IEEE 802.3   10/100 Ethernet  Up
36  ATM/0      ATM                   ATM              Up
37  ATM/1      ATM                   ATM              Up
38  ATM/2      ATM                   ATM              Down
39  ATM/3      ATM                   ATM              Down
40  Eth/32     Ethernet/IEEE 802.3   ATM              Up
41  Eth/33     Ethernet/IEEE 802.3   ATM              Up
42  Eth/34     Ethernet/IEEE 802.3   ATM              Up
43  Eth/35     Ethernet/IEEE 802.3   ATM              Up
44  Eth/36     Ethernet/IEEE 802.3   ATM              Up
45  Eth/37     Ethernet/IEEE 802.3   ATM              Up
46  Eth/38     Ethernet/IEEE 802.3   ATM              Up
```

```
47  Eth/39     Ethernet/IEEE 802.3   ATM             Up
48  Eth/40     Ethernet/IEEE 802.3   ATM             Up
49  Eth/41     Ethernet/IEEE 802.3   ATM             Up

50  Eth/42     Ethernet/IEEE 802.3   ATM             Up
51  Eth/43     Ethernet/IEEE 802.3   ATM             Up
52  Eth/44     Ethernet/IEEE 802.3   ATM             Up
53  Eth/45     Ethernet/IEEE 802.3   ATM             Up
54  Eth/46     Ethernet/IEEE 802.3   ATM             Up
55  Eth/47     Ethernet/IEEE 802.3   ATM             Up
56  Eth/48     Ethernet/IEEE 802.3   ATM             Up
57  Eth/49     Ethernet/IEEE 802.3   ATM             Up
58  Eth/50     Ethernet/IEEE 802.3   ATM             Up
59  Eth/51     Ethernet/IEEE 802.3   ATM             Up
60  Eth/52     Ethernet/IEEE 802.3   ATM             Up
61  Eth/53     Ethernet/IEEE 802.3   ATM             Up
62  Eth/54     Ethernet/IEEE 802.3   ATM             Up
63  Eth/55     Ethernet/IEEE 802.3   ATM             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 SNMP protocol console process.

   ```
   + protocol 11
   ```

   or

   ```
   + protocol SNMP
   ```

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

   ```
   SNMP>
   ```

You can now begin entering the protocol's commands. See the corresponding protocol section of the *Protocols and Features* for more information on specific protocol console commands.

Command Completion

The automatic command completion function assists you with the syntax for commands entered at the command line.

To illustrate the behavior of Command Completion, assume that the following commands are allowed in a given menu context. (This is an example menu only.)

enable
- auto-refresh
- caching

set
- cache-size
- cache-timeout
- priority

- If you type **ena** and press the Space Bar, the full command is shown as **ENABLE**. If you now type **?**, a list of possible items to enable are shown (**auto-refresh** and **caching**) and the command **ENABLE** remains on the command line.

- If you type **ena** and press **Enter**, a message is printed that the command is not fully specified, and a list of possible items to enable are shown (**auto-refresh** and **caching**) and the command **ENABLE** remains on the command line.
- Because the **ENABLE** command requires an item to enable, it appears in a list of possible command completions with "..." in the left margin to indicate that more input is required for the command.
- If your input matches multiple commands, a list of possible completions is displayed. Your input on the new command line is expanded to the longest common prefix. For example, if you enter **set ca**, and then press the space bar, **CACHE-SIZE** and **CACHE-TIMEOUT** will be listed, and the new command line will be expanded to **SET cache-**, since "cache-" is common to both possible completions. Now you must type the letter "s" or the letter "t" to distinguish between the possible completions ″size″ or ″timeout″.
- Common commands sometimes appear in an alternate form (**SHOW**, **DISPLAY**, **LIST**). If the Command Completion does not yield a match on a common command, such as **SHOW**, the alternatives **DISPLAY** or **LIST** will be displayed, if found.
- If the search for a command (and alternatives) does not yield an exact match, you are presented with a list of possible completions, using some portion of your input. For example, **enanle** followed by the Space Bar would be replaced with **ena** and **ENABLE** would be listed as the possible completion.
- When a list of possible commands is shown, you can use the Tab key to cycle through one command at a time on the current command line. You can use the Space Bar or Enter key to select the command shown.

Online Help When Command Completion is Enabled

The following online help is available when command-completion is enabled.

See page 61 for the **enable command-completion** syntax.

? Question mark displays a list of possible completions. A message appears if the command is already complete.

Space Bar
Attempts to complete the current word on the command line. If a unique match is not found, possible completions are listed.

Tab Attempts to complete the current word on the command line. If a unique match is not found, possible completions are listed and you may cycle through these possible completions using the Tab key. Use the Space Bar or the Enter key to select the currently displayed command.

Enter Attempts to complete the current word on the command line. If the command is complete, Enter executes the command and stores it in the Command History. If the command is incomplete, a list of possible completions is displayed.

Ctrl-P Returns to the MOS Operator Console prompt (*). (Ctrl-P is the default Intercept Character.)

Backspace
Deletes the last character on the command line.

Ctrl-W Deletes the last word on the command line.

Ctrl-U Aborts the current command.

Ctrl-L Refreshes the current command line to display its contents.

Ctrl-B Retrieve Backward. Replaces the current command line with the previous command in the circular Command History.

Ctrl-F Retrieve Forward. Replaces the current command line with the next command in the Command History.

Ctrl-R Marks the start of a Repeat Sequence in the Command History. Use with the **Ctrl-N** function.

Ctrl-N Replaces the current command line with the next command in the Repeat Sequence whose starting command was marked with **Ctrl-R**.

Ctrl-C Cancels Easy-Start, if active.

Escape ?
Escape, followed by "?" prints this Command Line Help:

The following rules apply to automatic command completion:

- Completed commands are shown in UPPERCASE on the command line.
- Common commands sometimes appear in an alternate form (**ADD** versus **CREATE**). If the command completion does not yield a match on a common command, any alternative commands will be displayed.
- If the search for a command (and alternative commands) does not yield a unique match, a list of possible completions is shown, and the longest common prefix is presented.
- When possible completions are listed, commands requiring further command input are shown with "..." in the left margin.
- When a Command History retrieve key (Ctrl-B,F,N) is pressed, the Command History is scanned for a command that successfully parses in the current command context. A tone will be sounded if no such command exists.
- Some command menus are built dynamically. Command Completion cannot always follow these dynamic links. '?' can be entered in these cases.
- To disable Command Completion for just one command (to enter a comment), type any Comment Character as the first character on the command line. The Comment Characters are !@#$%*:;/'"
- Command Completion will be disabled in the event of an internal error. Report the Debug information on the screen to Customer Support.
- Command Completion is currently Enabled. To Disable this option, use the **disable command-completion** command from Configuration `talk 6`.

Online Help When Command Completion is Disabled

The following online help is available when command-completion is disabled:

? When a ? (Question Mark) is entered at the end of the command line, a list of possible completions is shown.

Enter Executes the command and stores it in the Command History. A message is printed if the command is not fully specified

Ctrl-P Returns to the MOS Operator Console prompt (*). (Ctrl-P is the default Intercept Character.)

Backspace
Deletes the last character on the command line.

Ctrl-U Aborts the current command.

Ctrl-B Retrieve Backward. Replaces the current command line with the previous command in the circular Command History.

Ctrl-F Retrieve Forward. Replaces the current command line with the next command in the Command History.

Ctrl-R Marks the start of a Repeat Sequence in the Command History. Use with the **Ctrl-N** function.

Ctrl-N Replaces the current command line with the next command in the Repeat Sequence whose starting command was marked with **Ctrl-R**.

Ctrl-C Cancels Easy-Start, if active.

Escape ?
Escape, followed by "?" prints this Command Line Help:

```
Command Completion is currently Disabled.  To Enable this option,
use the enable command-completion command from Configuration talk 6.
```

Command History

The Command History contains up to the last 50 commands entered by the user in OPCON, 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 pressing **Ctrl-B** (backward) or **Ctrl-F** (forward) at any command line prompt in an OPCON, 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 across the command line interface. 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.

When automatic Command Completion is enabled (See"Command Completion" on page 16) and a Command History retrieve key (Ctrl-B,F,N) is pressed, the Command History is scanned for a command that successfully parses in the current command context. A tone will be sounded if no such command exists.

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 reload, pressing **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 tone sound.

Note: A command aborted by pressing **Ctrl-U** will not be entered into the Command History. When Command Completion is enabled, only complete commands are entered into the Command History.

To enter two similar commands:

```
display sub les
```

display sub lec

Enter:

display sub les, then press **Enter**

Ctrl-B for Backward, and the current line is replaced with-

display sub les

Press **Backspace** and replace “s” with “c” to get

display sub lec and 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 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.

   ```
   *console
   +event Ctrl-R
   ```

   then press **Enter** to set the start of the repeat sequence.
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-
*configuration
Config>Ctrl-N for NEXT to retrieve the start of this sequence-
Config>event Enter
Event Logging System user configuration
ELS config>Ctrl-N for NEXT to retrieve the next command in sequence-
ELS config>display sub les Enter
ELS config>Ctrl-N for NEXT to retrieve the next command in sequence-
ELS config>display sub lec Enter
ELS config>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:

```
*console
+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.

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

Chapter 3. Using Service Functions in the IBM 8371 and 8371 Blade Firmware

This section covers boot options that can be set from the Firmware bootstrap menus. For information about file transfer and file management, refer to *Networking Multilayer Ethernet Switch Installation and Planning Guide*.

The purpose of the bootstrap firmware is to provide a Power On Self Test (POST) for the IBM 8371 processor card and to boot from the active operating system image stored in the FLASH memory.

Two operating system images are stored in system FLASH. The active boot image is selected by using the `Configuration` **boot** command. The active image selected using the **boot** command is used to boot the device. You can select the backup boot image or the Ethernet service port using commands available from the firmware bootstrap menus. See “Select Boot Device” on page 26.

Accessing the Firmware Bootstrap Menus

Before booting the device, note that:

- You will need a terminal or terminal emulator connected to the 8371 RS232 service port with a line speed of 19200 baud. This can be a VT100 TTY device connected directly through the service port.

To display the bootstrap main menu, power on the 8371, and press **Ctrl-C** on the terminal keyboard after one second.

Important: To access the Firmware prompt, you must stop the 8371 boot. To stop it, you must have a TTY console directly attached to the serial port. When the 8371 starts its boot sequence, press **Ctrl-C** from the console to interrupt the boot sequence.

From the Main Menu panel shown in Figure 3 on page 24, you can select one of five services. The following sections explain these services and provide instructions for going through the associated panels:

© Copyright IBM Corp. 1996, 1999

```
8371 System Bootloader
VERSION: 1.00
(C) Copyright IBM Corporation, 1999 All Rights Reserved.


                       Bootstrap Main Menu

      1) Bootstrap Utilities
      2) Select Boot Mode
      3) Select POST Mode
      4) Select Boot Device

      9) Issue Reset

   Enter option:
```

Figure 3. Main Menu Panel

Bootstrap Utilities

The following options are available from the Utilities menu:

```
                    Bootstrap Utility Menu

      1) Display Error Log
      2) Clear Error Log

      8) Return to Bootstrap Main Menu
      9) Issue Reset


             Enter option:
```

Figure 4. Utilities Menu Panel

The utilities provide the following functions:

Table 2. Utilities

Function	Description
Display Error Log	Displays the self-test error log. See “Self-test Error Codes” on page 27 for a listing and description of the error codes.
Clear Error Log	Clears the self-test error log.
Return to Bootstrap Main Menu	Displays main menu as shown in Figure 3.
Issue Reset	Resets the hardware.

Select Boot Mode

The following options are available from the Select Boot Mode menu:

```
              Boot Sequence Selection Menu


      1) Normal Boot
      2) Service Boot
      3) Bootstrap Menu

      8) Return to Bootstrap Main Menu
      9) Issue Reset

      Current Setting : Normal Boot


              Enter option:
```

Figure 5. Select Boot Mode Menu Panel

Table 3 contains a description of the Boot Mode menu functions.

Table 3. Select Boot Mode Functions

Function	**Description**
Normal Boot	Boot and execute operating environment.
Service Boot	Boot and execute service environment.
Bootstrap Menu	Display Bootstrap menu. Does not cause a boot.
Return to Main Menu	Displays main menu as shown in Figure 3 on page 24.
Issue Reset	Resets the hardware.

Once selected, the IBM 8371 will stay in the selected boot mode until a different mode is selected from this menu.

Select POST Mode

The following options are available from the Select Post Mode menu:

```
                Select Post Mode Menu

      1) Set POST ON (Next Hard Reset only)
      2) Set POST OFF

      8) Return to Bootstrap Main Menu
      9) Issue Reset

      Current Setting :  OFF


              Enter option:
```

Figure 6. Select Post Mode Menu Panel

Table 4 contains a description of the Boot Mode menu functions.

Table 4. Select POST Mode Functions

Function	Description
Set POST on (next reset only)	This option causes a more extensive self-test (POST) to be executed the next time the device is reset or powered on. Once the extended POST executes successfully, the normal, short POST will be executed on successive executions of POST.
Set POST off	Indicates that the normal, short POST will be executed.
Return to Bootstrap Main Menu	Displays main menu as shown in Figure 3 on page 24.
Issue Reset	Resets the hardware.

Select Boot Device

The following options are available from the Select Boot Device menu:

```
                Select Boot Device Menu


       1) Boot from active Flash bank
       2) Boot from backup Flash bank
       3) Boot from Ethernet service port
       4) ZMODEM boot via RS 232 service port

       8) Return to Bootstrap Main Menu
       9) Issue Reset

       Current Setting : Active Flash Bank


              Enter option:
```

Figure 7. Select Boot Device Menu Panel

Table 5 contains a description of the Boot Device menu functions.

Table 5. Select Boot Device Functions

Function	Description
Boot from active Flash bank (normal boot)	Two copies of the operational code exist in FLASH. The active image is the image from which a normal boot occurs.
Boot from backup Flash bank (backup boot)	Indicates that the next boot only of the IBM 8371 is to be from the alternate (backup) image. The alternate image is intended for recovery when the active image has become corrupted and cannot be booted. Once the IBM 8371 has been booted, the boot device will be automatically reset to the active Flash bank.

Table 5. Select Boot Device Functions (continued)

Function	Description
Boot from Ethernet service port	Selects TFTP boot from an Ethernet server with IP address 10.1.2.3 the next time the IBM 8371 is booted. The filename on the server must be os8371.ld. The file must be located in the TFTP server directory on the workstation. On an AIX workstation, the default directory is /tftpboot. TFTP applications on Windows workstations allow you to designate the TFTP directory. If you do not have a TFTP application, you can download one from the Web at **www.alphaWorks.ibm.com/formula/tftp**. Once the IBM 8371 has been booted, the boot device will be automatically reset to the active Flash bank.
ZMODEM boot via RS 232 service port	Selects ZMODEM boot from a workstation attached to the RS 232 service port the next time the IBM 8371 is booted. The boot image must be stored on the workstation and the workstation must support ZMODEM sendfile. The file transfer rate is fixed at 19.2 Kbps and the image download will take about 36 minutes. This function supports boot image recovery if both images in the FLASH memory are corrupted and TFTP boot via the Ethernet service port is not possible. Once the IBM 8371 has been booted, the boot device will be automatically reset to the active Flash bank.
Return to Main Menu	Displays main menu as shown in Figure 3 on page 24.
Issue Reset	Resets the hardware.

Issue a Hardware Reset

This function resets the IBM 8371 and executes the selected option. This reset causes a POST to execute.

Self-test Error Codes

The following error codes are displayed on the service terminal and logged in the self-test error log when self-test detects a hardware failure. Select option 1 on the Bootstrap Utility Menu (see “Bootstrap Utilities” on page 24) to display the error log.

Table 6. Self-test Error Codes

Error Code	Physical Location	Explanation
00010106	CPU Card	Processor Dual Port RAM Failure (fatal)

Table 6. Self-test Error Codes (continued)

Error Code	Physical Location	Explanation
00010206	CPU Card	Processor Register Read/Write Data Miscompare
00040101	CPU Card	SCC2 UART No Transmit
00040102	CPU Card	SCC2 UART No Receive
00040106	CPU Card	SCC2 UART Wrap Data Miscompare
00060201	CPU Card	CPU I2C EEPROM write failure
00060202	CPU Card	CPU I2C EEPROM Read address command failure
00060203	CPU Card	CPU I2C EEPROM Read command failed
00060204	CPU Card	CPU I2C EEPROM Read receive failed
00070101	CPU Card	On Card Flash Status Error
00070102	CPU Card	On Card Flash Write Operaton Failure
00070103	CPU Card	On Card Erase Operation Failure
00070104	CPU Card	On Card Flash ID Error
00070105	CPU Card	On Card Flash Erase Verify Failure
00070106	CPU Card	On Card Flash Read/Compare Failure
00080106	CPU Card	SIL Bridge Chip Read/Write Data Miscompare
01000202	SDRAM SE	SDRAM EEPROM read address command failure
01000203	SDRAM SE	SDRAM EEPROM read command failure
01000204	SDRAM SE	SDRAM EEPROM read receive failure
01000206	SDRAM SE	Invalid row+column read from SDRAM EEPROM
01000207	SDRAM SE	Invalid # banks read from SDRAM EEPROM
01030106	SDRAM CS3	Data Storage Read/Write Data Miscompare (pattern test)
01030206	SDRAM CS3	Data Line Read/Write Data Miscompare (walking bit test)
01030306	SDRAM CS3	Address Line Read/Write Data Miscompare (address test)
01030406	SDRAM CS3	Clear memory failed
01040106	SDRAM CS3	Data Storage Read/Write Data Miscompare (pattern test)

Chapter 4. Getting Started with Configuring the 8371

The 8371 is a plug-and-play device that boots with a default configuration. All possible interfaces are automatically configured at boot time. The following table shows the network interfaces available on the 8371.

Network Interfaces on the 8371

Table 7. Network Interfaces Automatically Configured on the 8371

Device Type	Slot	Ports	Interface Net Numbers
10/100 Ethernet (fixed ports on base switch)	0	1 - 16	0 - 15
10/100 Ethernet Feature Card	1	1 - 8	16 - 23
10/100 Ethernet Feature Card	2	1 - 8	24 - 31
LAG	3	1 - 4	32 - 35
ATM	1	1 - 2	36 - 37
ATM	2	1 - 2	38-39
Ethernet LAN Emulation Client	3	5 - 29	40 - 63

Note: All of the above interfaces cannot be active at the same time. For example, if Ethernet Feature cards are installed in both slots 1 and 2, there is no place to install ATM interfaces.

In addition to the network interfaces, the transparent bridge is also automatically configured. All the Ethernet interfaces and all of the LECs are configured as ports on the transparent bridge. Interfaces that are configured but are disabled or not actually present, are inactive bridge ports.

Network Interfaces on the IBM 8371 Blades

Table 8. Network Interfaces Automatically Configured on the 8265 Blade

Device Type	Slot	Ports	Interface Net Numbers
10/100 Ethernet (fixed ports on base switch)	0	1 - 16	0 - 15
10/100 Ethernet Feature Card	1	1 - 8	16 - 23
Reserved	2	1 - 8	24 - 31
LAG	3	1 - 4	32 - 35
Reserved	1	1	36 - 37
ATM (connection to 8265 backplane)	2	1	38
Reserved	2	1 - 2	39
Ethernet LAN Emulation Client	3	5 - 29	40 - 63

Table 9. Network Interfaces Automatically Configured on the 8260 Blade

Device Type	Slot	Ports	Interface Net Numbers
10/100 Ethernet (fixed ports on base switch)	0	1 - 16	0 - 15

© Copyright IBM Corp. 1996, 1999

Table 9. Network Interfaces Automatically Configured on the 8260 Blade (continued)

Device Type	Slot	Ports	Interface Net Numbers
10/100 Ethernet Feature Card	1	1 - 8	16 - 23
Reserved	2	1 - 8	24 - 31
LAG	3	1 - 4	32 - 35
ATM	1	1-2	36-37
Reserved	2	1 - 2	38-39
Ethernet LAN Emulation Client	3	5 - 29	40 - 63

LEC Configuration Details

Only one of the 24 automatically Ethernet LEC interfaces is enabled by default. The LEC with an interface number of 40 is enabled, while LECs with interface numbers of 41 - 63 are disabled. The configured LEC has:

ATM interface 36

ELAN name ELAN*j*, where *j* is the interface number of the LEC and *j*= (interface number of the LEC – 39)

LECS Auto-config yes

MAC Address/ESI Burned-in MAC address for net *i*

Selector 0

This chapter explains how to access the 8371 using a workstation and how to manage operational software images and configuration files. It also provides a brief overview of the configuration methods available for the 8371.

Configuration and Monitoring Tools

These are the various configuration and monitoring tools that are supported by the physical connections:

Web browser Hypertext Markup Language (HTML) interface

The Web browser interface is a configurator that is a home page and is accessed by a Web browser from a workstation that is connected to the 8371. You need a Web browser that can display clickable images and tables. The Web browser interface can be accessed using SLIP or IP. You must use the serial line connection and SLIP before the 8371 is operational in the network.

If you supply the Web browser the SLIP address, one of the configured IP addresses of the 8371, or its name (when using an IP name server), the Web browser interface will come up.

Note: The configured IP addresses of the 8371 include the IP addresses of all the LAN emulation clients .

Command line interface

The command line interface is a teletypewriter (TTY) text interface that requires you to enter commands to use it. The workstation that accesses it must be either an ASCII terminal, a personal computer (PC), or other intelligent programmable workstation emulating an ASCII terminal.

This interface must be reached over a serial link before the 8371 is operational in the network; you can use TTY or SLIP to access it. If you use SLIP, you can Telnet into the 8371.

After the 8371 is operational in the network, you can Telnet into the 8371 over IP to bring up this interface. If one connection to the 8371 is a Telnet session, the 8371 can support two connections at one time.

The command line interface is marked by an asterisk (*) prompt.

Important: If you use a serial connection, (either local or remote), you ***must press a key*** to bring up the asterisk that is the prompt for the command line interface. When you make the connection, the message `Please press a key to obtain console` appears and reminds you to do this.

Local and Remote Console Access

When accessing the 8371 locally on a null modem cable attached to the EIA service port, use VT100 terminal emulation. Because VT100 does not define function keys above F4, edit the keyboard mapping manually as follows: For F6, enter the mapping (ESC)OU. For F9, enter the mapping (ESC)(Left square bracket)009q.

Note: (ESC) represents the carat symbol followed by the left square bracket.

File Transfer

Table 10 defines the ways in which configuration files and operational software files can be transferred to and from the 8371.

Table 10. File Transfer

File Transfer Method	Type of Connection
TFTP Get command from the 8371 to the workstation that has the binary configuration file, to download operational software images and configuration files to the 8371. Files sent using TFTP must be binary, or the 8371 cannot use them. You should use the Create command of the Configuration Program to save configuration files in binary format before downloading them to the 8371. After the 8371 is operational in the network, you can use the TFTP Put command over IP to upload a file from the 8371 to a workstation. The file will be uploaded in binary format. Both operational software and configuration files can be uploaded. You should use the Read device configuration option of the Configuration Program to make an uploaded configuration file usable by the Configuration Program so that you can change some parameter values in it. **Note:** Using TFTP Put is a way to retrieve files from the 8371 if the Retrieve option of the Configuration Program is not available.	• SLIP connection (using the TFTP Get command to download files to the 8371). • IP connection of operational 8371 over functioning network (using the TFTP Get and Put commands to download and upload files).

Table 10. File Transfer (continued)

File Transfer Method	Type of Connection
The Communications Option of the Configuration Program (actually, the protocol for this is SNMP). This method cannot be used until the 8371 is operational in the network. The files are not binary, but are in a format that is internal to the Configuration Program. This function can send configuration files to the 8371 and retrieve them from the server.	IP connection of operational 8371 over functioning network.

Tips for Managing Configuration Problems

Important: After the IBM 8371 is configured and operational, ***always*** back up the active configuration file. Keeping this file enables you to re-establish the IBM 8371 on the network should the active configuration become corrupted.

Back up the active configuration file by retrieving it and storing it in the workstation. See “File Transfer” on page 31 for more information.

Reconfiguring

You may find it hard to detect problems caused by configuration errors. A configuration error can initially appear to be a hardware problem because the IBM 8371 will not start or data will not flow through a port. In addition, problems with configuration may not result in an error initially; an error may occur only when specific conditions are encountered or when heavy network traffic occurs.

If you cannot resolve a problem after making a few changes to the configuration or after restoring the active configuration file, it is recommended that you generate a new configuration. Too many changes to a configuration often compound the problem, whereas you can usually generate and test a new configuration within a few hours.

How Software Files Are Managed

To help manage operational software upgrades and configurations, the IBM 8371 has a software change management feature. This utility enables you to determine which operational software file and which configuration file is active while the IBM 8371 is running. In addition to storing the active operational software and the active configuration file, the IBM 8371 stores two backup images of the operational software and up to 4 configuration files in non-volatile memory.

How to View the Files

To use the change management tool in the command line interface to view the operational software image and the configuration files, follow these steps:

1. From the prompt for OPCON, which is an asterisk (*), type **talk 6**. The prompt `Config>` appears.
2. Enter **boot**. You will see the prompt `Boot config>`.
3. Enter **list** to display information about which load images and configuration files are available and active.

   See “List” on page 78 for sample list output and a description of file statuses.

How to Reset the IBM 8371

Note: A reset interrupts the function of the IBM 8371 for up to 90 seconds. Be sure that the network is prepared for the interruption.

As previously stated, PENDING and LOCAL files are not loaded into active memory until you reset the IBM 8371.

To reset the IBM 8371, type **reload** at the OPCON prompt (*) .

File Transfer Using TFTP

See “TFTP” on page 80 for a sequence of commands to transfer a file from a workstation or server to the IBM 8371 using TFTP. You will need to substitute your own values for the IP address and path, which are given as examples.

Storing Configuration Files Using the Command Line Interface or the Web Browser Interface

To store a configuration file created using the command line interface, type **write** at the `Config>` prompt. When using the Web browser interface, select **Write**. The Write command creates a binary configuration file that contains the most current value of each of the configuration parameters.

This file is stored in the ACTIVE bank and is given PENDING status. If the status of the file is not changed by a Set command, it becomes the ACTIVE configuration when the IBM 8371 is reset.

Changing the Statuses of Files

These are the ways to change the statuses of image and configuration files:

- You can cause the IBM 8371 to perform a reset by using the Send command from the Communications Option of the Configuration Program. When you do this, the file sent can arrive as a PENDING file or as an AVAIL file. If it is a PENDING file, it becomes the ACTIVE configuration and the previously ACTIVE file becomes AVAIL when the IBM 8371 is reset.

  If it is an AVAIL file, resetting the IBM 8371 does not change its status.
- Use the Set config (set config) commands from the `Boot config>` prompt manually to change the status of any files except the ACTIVE files. If you set a file to PENDING, it becomes ACTIVE and the ACTIVE file becomes AVAIL when a reset is performed.
- Use the Write command to store a configuration file that you have created using the command line interface or the Web browser interface, it is stored with a PENDING status.
- If you copy a file from one location to another, the file receives the status of the file that was there before it and that it overwrites. For example, copying a file with the status of AVAIL over a file that has the status of PENDING, the new file will keep the status of the original file, which is PENDING.

Using the Set Commands

See “Set” on page 79 for information about the **set** command.

Other Change Management Functions

These are the other change management commands:

- Describe load images
- Describe config images
- Disable dumping
- Enable dumping
- Erase files

Describe

See “Describe” on page 77 for information about the **Describe** function.

Enable Dumping

This command enables the dumping of memory without intervention from anyone in the event that the IBM 8371 has a catastrophic error. The IBM 8371 will dump memory onto the hard disk. Once a successful dump has been taken, the IBM 8371 attempts to restart. Depending upon the failure of the IBM 8371, it cannot always restart. In this case, you should restart it manually and call a service person, who will dial into the IBM 8371 to determine the nature and the causes of the failure.

To enable dumping, type **t 6** at the *, press **Enter** and then type **enable dump** or **ena du** at the `Config>` prompt. You will see the message:

```
Config> Automatic memory dump enabled
```

The default state is to have dumping enabled.

Erase Files

See “Erase” on page 77 for information about the **erase** command.

Using the Copy Command

The Copy command moves a file from one location in the storage area to another. This command allows you to change the status as well. The file moved always receives the status of the storage area that it is moved to. For example, suppose that you have this scenario:

- The configuration file in BANK A CONFIG 1 is AVAIL. The configuration file in BANK B CONFIG 1 is PENDING.
- You copy the configuration in BANK A CONFIG 1 to BANK B CONFIG 1.

In this case, the original configuration file in BANK A CONFIG 1 remains unchanged and AVAIL. The configuration that was in BANK B CONFIG 1 is overwritten by a copy of the configuration file that is in BANK A CONFIG 1. This copy retains the status of the file that it overwrote, in this case, PENDING.

See “Copy” on page 76 for additional information about the **copy** command.

Using the Lock Command

The **lock** command prevents the device from overwriting the selected configuration with any other configuration.

See “Lock” on page 79 for additional information about the **lock** command.

Using the Unlock Command

The **unlock** command removes the lock from a configuration allowing the configuration to be updated.

See “Unlock” on page 80 for additional information about the **unlock** command.

Chapter 5. Using the World Wide Web Interface

The IBM 8371 provides a World Wide Web interface to monitor and configure the product. The Web browser interface performs all of the functionality of the command line interface, but in a graphical, more user-friendly manner.

Connecting to the World Wide Web Interface

Use any Web browser that supports HyperText Markup Language (HTML) tables and clickable images. Examples of browsers that support this feature are WebExplorer Version 1.03 or higher, Netscape Navigator Version 1.1N or higher, and Mosaic Version 2.1.1 or higher.

Access the Web interface through TCP/IP Host Services on the bridged network to which the IBM 8371 is connected or by using IP routing.

You will be shown the Home Page that is described in the next section.

Rules for Using the Web Interface

When configuring using the Web browser interface, observe the following guidelines:

- Many configuration options require you to enter data on two or more Web pages (or forms). If you fill in and submit the first form in a series, be sure to complete the remaining forms. If you do not fill in and submit all the forms, the configuration parameter could be left in an unknown state.
- More than one person should not perform configuration at the same time. They can interfere with one another. For example, one person could delete an interface while the other person is in the middle of configuring a protocol on that interface.
- Disable the caching feature of the browser. If you do not do this, the browser may pull a page out of memory instead of going to the IBM 8371 to get the latest information. The browser will display old data. This problem is more likely to occur when you use the *Back* button on the browser.
- Do not use your Web browser's reload, back, or forward navigation buttons when using the Web browser interface. Using these buttons could cause problems during configuration. Instead, use the command history list or any of the navigation buttons on the Web pages themselves.

Home Page Structure

The Home Page provides a graphic that shows the status of the IBM 8371. It indicates the current network interfaces installed and shows the status of each port (for example, installed, enabled, or disabled). The current state of each LED is also shown, as well as the indication of the devices that are installed.. If the Web browser supports dynamic refresh, then this page will automatically refresh itself approximately every 80 seconds. If you click any of these ports or interfaces, a more detailed description of its status will be shown on a separate Web page.

Click **How to use this Web Site** for instructions about using this site.

Click **Configuration and Console** to bring up the menu.

© Copyright IBM Corp. 1996, 1999

Click **Diagnostics** to bring up the menu shown in Figure Figure 8.

Click **Vital Product Data** for information about the hardware and operational software. This panel, which is usually used for diagnostics, is not displayed here.

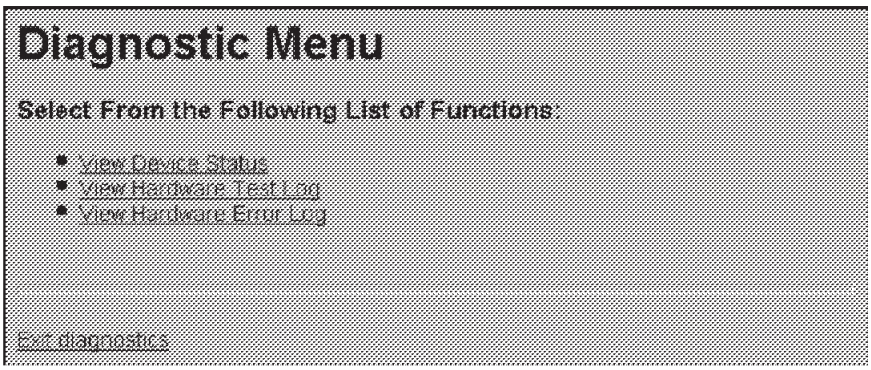

Figure 8. Diagnostic Menu

Event Logging System

One of the links on the Configuration and Console page 1 is to the Event Logging System (ELS). The ELS display is similar to the one provided on the command line interface. On the Web interface, going into the ELS will display the most recent events stored in the system memory. In order to get future updates, press the Reload button on your browser. For more details about the ELS message facility, refer to the *Event Logging System Messages Guide*.

Operator Console

The console monitoring interface provides real-time status information very similar to that offered in the command line interface. The menus from the command line interface are presented as a hierarchy of Web links that can easily be traversed with the click of a mouse button. It is possible to jump back several levels in the hierarchy with a single push of a button.

Device Configuration

Important: Exercise caution when using the Web browser to change configuration parameters. Changes to the configuration that are made using the Web browser are written directly to static random access memory (SRAM). You can make unintentional configuration changes that do not take effect until you reset the IBM 8371. To check that you have the correct parameters, look over the settings for any parameters that you have configured before submitting them.

The Web interface greatly simplifies the configuration of network and protocol parameters. In many cases where it is necessary to remember the individual network numbers on the command line interface, those options are now all presented as menu options on the Web. Also, the Web interface uses the graphical features available to it, such as pick lists, selection lists, radio buttons, and check boxes.

If a particular configuration option needs to prompt you for answers to several questions, those questions will be presented on a single Web page. After all of the questions are filled in, you should press the *Submit* button to send the data back to the IBM 8371 for validation.

The hierarchy of the Web browser interface is very similar to that of the command line interface.

History Function

The Web Configurator uses a selection list and a *Return to* button to provide an advanced history function. Depending upon your choice of HTML browser, a pick list, choice box or pull-down list box will be displayed. This list of selections contains the names of the pages visited under the current branch of the software structure. To return to a previously visited page within the current command pathway, select that entry from the list and click the *Return To* button.

Chapter 6. The OPCON Process and Commands

This chapter describes the OPCON interface configuration and operational commands. It includes the following sections:

- "What is the OPCON Process?"
- "Accessing the OPCON Process"
- "OPCON Commands"

What is the OPCON Process?

The Operator Console process (OPCON) is the root-level process of the device software user interface. The main function of OPCON is to communicate with processes at the secondary level, such as Configuration, Console, and Event Logging. Using OPCON commands, you may also:

- Display information about device memory usage
- Reload the device software (reboot)
- Telnet or ping to other devices or hosts
- Display status information about all device processes
- Manipulate the output from a process
- Change the OPCON intercept character

Accessing the OPCON Process

When the device starts for the first time, a boot message appears on the console. Then the OPCON prompt (*) appears on the console, indicating that the OPCON process is active and ready to accept commands.

The OPCON process allows you to configure, change, and monitor all of the device's operating parameters. While in the OPCON process, the device is forwarding data traffic. When the device 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 device's operating parameters made while in OPCON take effect immediately without requiring reinitializing of the device. If the changes do not take effect, use the **reload** command at the * prompt.

At the * prompt, an extensive set of commands enables you to check the status of various internal software processes, monitor the performance of the device's interfaces and packet forwarders, and configure various operational parameters.

OPCON Commands

This section describes the OPCON commands. Commands that are needed more often appear before the "- - - - -" separator. Each command includes a description, syntax requirements, and an example. The OPCON commands are summarized in Table 11 on page 42. To use them, access the OPCON process and enter the appropriate command at the OPCON prompt (*).

© Copyright IBM Corp. 1996, 1999

Table 11. OPCON Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Configuration*	Accesses the device’s configuration process. (`talk 6`)
Console*	Accesses the device’s console process. (`talk 5`)
Event Logging System*	Accesses the device’s event logging process. (`talk 2`)
ELS Console*	Accesses the device’s secondary ELS Console process. (`talk 7`)
Logout	Logs off a remote console.
Ping	Pings a specified IP address.
Reload	Reloads the device.
Telnet	Connects to another device.
- - - - - -	- -
Diags	Displays device status and the contents of the hardware test log and the hardware error log.
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.
Memory	Reports the device’s memory usage.
Status	Shows information about all device processes.
Suspend	Temporarily disables Command Completion for the current session only.
Talk	Connects to another device process and enables the use of its commands.

* When you use this command for the first time, you will be reminded that you can use **Ctrl-P** to return to the MOS Operator Console prompt (*).

Configuration

Use the **configuration** command to access the device’s configuration process (`talk 6`). See “Chapter 7. The CONFIG Process (CONFIG - Talk 6) and Commands” on page 53 for more information.

Syntax:

configuration

Example:

```
* configuration

(To return to the MOS Operator Console prompt (*), press Control-P)

Gateway user configuration
Config>
```

Console

Use the **console** command to access the device’s console and monitoring process (`talk 5`). See “Chapter 10. The Operating/Monitoring Process (GWCON - Talk 5) and Commands” on page 83 for more information.

Syntax:

console

Example:

```
* console

CGW Operator Console

+
```

Diags

Use the **diags** command to display the Diagnostic Main Menu. The diagnostic menus allow you to enable, disable and test hardware adapters or ports. Diagnostic menus have on-screen help for the various options and status information that is available.

You can use the “b” (back) key to return to any previous menu. Use the “e” (exit) key to exit the diagnostics and return to the OPCON command prompt.

Syntax:

diags

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 device allows only certain processes to be redirected.

The **divert** command requires 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 TTY0 after executing the **divert 2 0** command. If you want to stop the output, enter **halt 2**. The **halt** command is described in “Halt” on page 44.

Syntax:

divert *pid tty#*

Example:

```
* divert 2 0
* status
Pid  Name      Status TTY  Comments
1    COpCon    IDL    TTY0
2    Monitr    IDL    TTY0
3    Tasker    RDY     --
4    MOSDBG    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    ELSCon    DET     --
8    ROpCon    IDL    TTY1
9    ROpCon    RDY    TTY2 jlg@128.185.40.40
10   WEBCon    IDL    --
```

Els

Use the **els** command to access the device's secondary ELS console process, (`talk 7`). See "Accessing the Secondary ELS Console Process, ELSCon (Talk 7)" on page 13 for more information.

Syntax:

els

Event

Use the **event** command to access the device's event logging process, (`talk 2`). See "Chapter 12. Using the Event Logging System (ELS)" on page 97 for more information.

Syntax:

event

Flush

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

The device allows only certain processes to be flushed. To obtain the PID and tty#, use the OPCON **status** command. In the following example, after executing the **flush 2** command, the output of the MONITR process is sent to the Sink (it has been flushed).

Syntax:

flush *pid*

Example:

```
* flush 2
* status
Pid  Name      Status TTY  Comments
1    COpCon    IDL    TTY0
2    Monitr    IDL    Sink
3    Tasker    RDY     --
4    MOSDBG    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    ELSCon    DET     --
8    ROpCon    IDL    TTY1
9    ROpCon    RDY    TTY2
10   WEBCon    IDL    --
*
```

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 device 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. In

the following example, after executing the **halt 2** command, the MONITR process is no longer connected to TTY0. Event messages no longer appear.

Syntax:

halt *pid*

Example:

```
* halt 2
* status
Pid  Name      Status TTY  Comments
1    COpCon    IDL    TTY0
2    Monitr    IDL     --
3    Tasker    RDY     --
4    MOSDBG    DET     --
5    CGWCon    DET     --
6    Config    DET     --
7    ELSCon    DET     --
8    ROpCon    IDL    TTY1
9    ROpCon    RDY    TTY2
10   WEBCon    IDL    --
```

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 can be a control character. Enter the ^ (shift 6) character followed by the letter character or non-alphanumeric character, such as !@#$%, you want for the intercept character.

Note: This change applies to only the current login session.

Syntax:

intercept ^ *character*

Example 1:

```
* intercept ^a
```

From this example, the intercept character is now **Ctrl-A**.

Example 2:

```
* intercept !
```

From this example, the intercept character is now **!**.

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

Memory

Use the **memory** command to obtain and display information about the device's global heap memory usage. The display helps you to determine if the device is being utilized efficiently. For an example of memory utilization, see Figure 9.

See "Memory" on page 90 for memory usage via talk 5.

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.

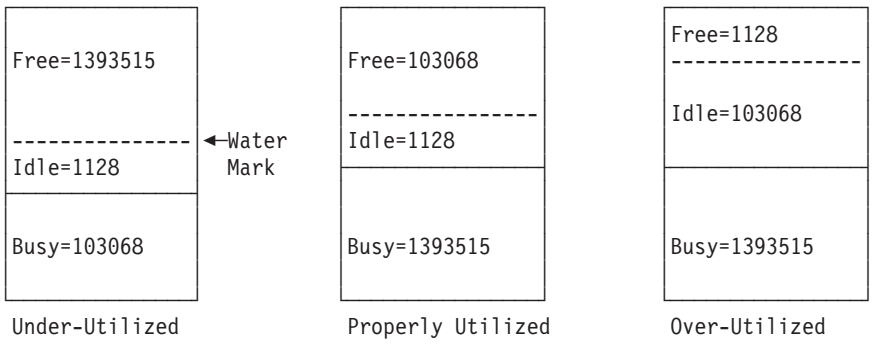

Figure 9. Memory Utilization

Ping

Use the **ping** command to have the device send ICMP Echo messages to a given destination (that is, "pinging") and watch for a response. This command can be used to isolate trouble in the internetwork.

Syntax:

ping *dest-addr [src-addr data-size ttl rate tos data-value]*

The ping process is done continuously, incrementing the ICMP sequence number with each additional packet. Each matching received ICMP Echo response is reported with its sequence number and the round-trip time. The granularity (time resolution) of the round-trip time calculation is usually around 20 milliseconds, depending on the platform.

To stop the ping process, type any character at the console. At that time, a summary of packet loss, round-trip time, and number of unreachable ICMP destinations will be displayed.

When a broadcast or multicast address is given as destination, there may be multiple responses printed for each packet sent, one for each group member. Each returned response is displayed with the source address of the responder.

You can specify the size of the ping (number of data bytes in the ICMP message, excluding the ICMP header), value of the data, time-to-live (TTL) value, rate of pinging, and TOS bits to set. You can also specify the source IP address. If you do not specify the source IP address, the device uses its local address on the outgoing interface to the specified destination. If you are validating connectivity from any of the device's other interfaces to the destination, enter the IP address for that interface as the source address.

Only the destination parameter is required; all other parameters are optional. By default the size is 56 bytes, the TTL is 64, the rate is 1 ping per second, and the TOS setting is 0. The first 4 bytes of the ICMP data are used for a timestamp. By default the remaining data is a series of bytes with values that are incremented by 1, starting at X'04', and rolling over from X'FF' to X'00' (for example, X'04 05 06 07 . . . FC FD FE FF 00 01 02 03 . . .'). These values are incremented only when the default is used; if the data byte value is specified, all of the ICMP data (except for the first 4 bytes) is set to that value and that value is not incremented. For example, if you set the data byte value to X'FF', the ICMP data is a series of bytes with the value X'FF FF FF . . .'.

Example:

```
* ping
Destination IP address [0.0.0.0]? 192.9.200.1
Source IP address [192.9.200.77]?
Ping data size in bytes [56]?
Ping TTL [64]?
Ping rate in seconds [1]?
Ping TOS (00-FF) [0]? e0
Ping data byte value (00-FF) [ ]?
PING 192.9.200.77-> 192.9.200.1:56 data bytes,ttl=64,every 1 sec.
56 data bytes from 192.9.200.1:icmp_seq=0.ttl=255.time=0.ms
56 data bytes from 192.9.200.1:icmp_seq=1.ttl=255.time=0.ms
56 data bytes from 192.9.200.1:icmp_seq=2.ttl=255.time=0.ms


----192.9.200.1 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max=0/0/0 ms
```

Reload

Use the **reload** command to reboot the device by loading in a new copy of the device software. When you use this command from a remote console, you install a new software load without going to the device. This command executes the same functions as pressing the reset button except that the device will not dump (if so configured). Before the reload takes effect, you are prompted to confirm the reload. You are also prompted if you have not saved the configuration changes.

Syntax:

reload

Example:

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

Status

Use the **status** command to display information about all device processes. By entering the PID after the **status** command, you can 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    IDL    TTY0
2    Monitr    IDL     --
3    Tasker    RDY     --
4    MOSDBG    DET     --
5    CGWCon    IOW     --
6    Config    IOW     --
7    ELSCon    DET     --
8    ROpCon    IOW    TTY1
9    ROpCon    RDY    TTY2
10   WEBCon    IDL    --
```

Pid Specifies the PID. This is the process to talk to 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.

TTY*n* Specifies the output terminal, if any, to which the process is currently connected.

TTY0 Local console

TTY1 or TTY2
Telnet consoles.

Sink 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).

Suspend

Use the **suspend** command to temporarily disable Command Completion for the current session only. If you are using an automated script, you can issue **suspend yes** as the first command if you want to temporarily disable Command Completion.

For information about Command Completion, see "Command Completion" on page 16.

Syntax:

suspend

Talk

You can use the **configuration**, **console**, or **event** commands to connect to other processes, such as CONFIG, GWCON, or MONITR, or use the **talk** command. 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 processes.

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

CGW Operator Console

+
```

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

Telnet

Use the **telnet** command to remotely attach to another device or to a remote host. The only optional parameter is the terminal type that you want to emulate.

You can use the **telnet** command with IPv4 addresses.

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

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

Syntax:

telnet *ip-address terminal-type*

Example 1: `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 '^]'
```

When telnetting to a non-existent IP address, the device 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 device,

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

  Note: When using a VT100 terminal, do not press ← **Backspace** because it inserts invisible characters. 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 reenter 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 7. The CONFIG Process (CONFIG - Talk 6) and Commands

This chapter describes the CONFIG process configuration and operational commands. It includes the following sections:

What is CONFIG?

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

- Set or change various configuration parameters
- Enter the Boot CONFIG command mode
- List or update configuration information
- Enable or disable console login
- Communicate with third-level processes, including protocol environments

Note: Refer to the chapter "Migrating to a New Code Level" in *8371 Networking Multilayer Ethernet Switch Installation and Planning Guide* for information about migrating to a new code level.

CONFIG lets you display or change the configuration information stored in the device's nonvolatile configuration memory. Changes to system and protocol parameters do not take effect until you reload the device software. (For more information, refer to the OPCON **reload** command in "What is the OPCON Process?" on page 41).

Note: You must enter the **write** command to save the changes in the device's flash memory.

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 SNMP protocol is `SNMP config>`.

If you want to know the process and mode you are communicating with, press **Enter** 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 13 on page 57 for a list of the commands you can issue from the CONFIG process.

Automatic Configuration

When the switch is booted, the following interfaces are allocated:

Table 12. Interfaces Added at Boot Time

Slot	Port	Device Type	Interface Number
0	1-16	10/100MB Ethernet	0-15
1	1-8	10/100MB Ethernet	16-23

© Copyright IBM Corp. 1996, 1999

Table 12. Interfaces Added at Boot Time (continued)

Slot	Port	Device Type	Interface Number
2	1-8	10/100MB Ethernet	24-31
3	1-4	LAG	32-35
1	1-2	ATM*	36-37
2	1-2	ATM*	38-39
3	5-29	LEC	40-63

Note: * Only 2 ATM ports are supported at one time. You can have 4 independent configurations of ATM ports, but only 2 are active at any one time.

When a feature card is hot-swapped into the switch, interface numbers are assigned from the above table. Feature cards can be added and removed to and from the feature card slots. However, the card being swapped must have been present at boot time, and only the same type of card may be swapped with the one being removed.

Dynamic Activation of a LEC

When a LEC is activated, the LEC must be associated with an ATM interface. The following default configuration values are associated with the LEC**:

ELAN Name ELAN1

ESI Set to MAC address stored in flash memory for this interface value

LES ATM Address set to use autoconfig via the LECS

ATM Interface 36

ELAN Type Ethernet

Bridging Enabled

Selector 2

Note: **These values guarantee that the box comes up with a known configuration. However, attempting to configure a LEC using all default values will probably fail. You need to provide network-specific information when configuring the LEC. See “Chapter 22. Configuring and Monitoring LAN Emulation Clients” on page 191 for LEC configuration detail.

Quick Configuration

Quick Configuration (Quick Config) provides a minimal set of commands that allow you to configure bridging protocols and routing protocols present in the device load. 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.

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

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 Quick Config.

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 **reload** command as a response to some Quick Config questions.

Manual Entry Into Quick Config Mode

You might want to run Quick Config manually to demonstrate the device's capabilities or to reconfigure dynamically to perform benchmark tests without having to learn the device'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 device 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*.

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 58.

Resetting Interfaces

Occasionally, you might need to change the configuration of a network interface along with its bridging and routing protocols without restarting the device. The **reset** command allows you to disable a network interface and then enable it using new interface, bridging and routing configuration parameters.

The interface, protocols and features configuration parameters are changed using the CONFIG process (talk 6) commands. The talk 6 commands affect the contents

of the configuration memory. The configuration changes are activated by issuing the GWCON process (talk 5) **reset** command.

To reset an interface:

1. Access the CONFIG process (talk 6).
2. Use the **net** command and other commands to change configuration parameters.
3. Use the **protocol** and **feature** commands to change the interface-based configuration parameters.
4. Exit the CONFIG process by pressing **Ctrl-P**.
5. Access the GWCON process (talk 5).
6. Use the **reset** command to reset the interface and the protocols and features on the interface.

Example:

```
* configuration
Config> net 0
ATM User Configuration
ATM Config> le-client
ATM LAN Emulation Clients Configuration
LE Client config> config 6

. . . change ATM LAN Emulation Client parameters . . .

Ethernet Forum Compliant LEC Config> exit
LE Client config> exit
ATM Config> exit
```

Note: When using the configuration program, do the following to make configuration changes to existing interfaces:

1. Make the configuration changes for the interface on the device
2. Enter the **reset** command to reset interface, protocol and feature parameters
3. Retrieve the configuration using the configuration program
4. Save the retrieved configuration into the configuration program database

Entering and Exiting CONFIG

To enter the CONFIG process from OPCON and obtain the CONFIG prompt, enter the **configuration** command. Alternatively, you can enter the OPCON **talk** command and the PID for CONFIG. The PID for CONFIG is 6.

```
* configuration
```

or

```
* talk 6
```

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

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

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 13.

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

Table 13. CONFIG Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Adds a user to the device.
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. Forces a re-boot for re-autoconfig. See Table 7 on page 29.
Disable	Disables command completion, login from a remote console, system memory dumping and rebooting, or a specified interface.
Enable	Enables command completion, login from a remote console, system memory dumping and rebooting, or enables a specified interface.
Event	Enters the Event Logging System configuration environment.
Feature	Provides access to configuration commands for independent device features outside the usual protocol and network interface configuration processes.
List	Displays system parameters, hardware configuration, a complete user list.
Network	Enters the configuration environment of the specified network.
Patch	Modifies the device’s global configuration.
Performance	Provides a snapshot of the main processor utilization statistics.
Protocol	Enters the command environment of the specified protocol.
Set	Sets system-wide parameters for buffers, host name, inactivity timer, packet size, prompt level, number of spare interfaces, dump parameters, location, and contact person.
Time	Keeps track of system time and displays it on the console.
Unpatch	Restores patch variables to default values.

Add

Use the **add** command to add user-access.

Syntax:

add user . . .

user *user_name*

Gives a user access to the device. You can authorize up to 50 users to access the device. 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 the table below.

When users are added, set login authentication to local. Otherwise a remote server must be used.

Table 14. Access Permission

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

Boot

Use the **boot** command to enter the Boot CONFIG command environment. For Boot CONFIG information, see “Chapter 8. Using BOOT Config to Perform Change Management” on page 73.

Syntax:

boot

Change

Use the **change** command to change your own password, or change user information.

Syntax:

change user

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 device's configuration information from nonvolatile configuration memory.

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

Syntax:

clear

- all
- arp
- asrt
- atm (Asynchronous Transfer Mode)
- bgp
- boot
- device
- els (Event Logging System Information)
- hostname
- ip
- ipx
- mpoa
- ospf
- prompt
- qos
- snmp
- tcp/ip-host
- time (Time of day information)
- user
- vlan

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. 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 device 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.

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 clearing.

Delete

Use the **delete** command to remove a user. To use the **delete** command, you must have administrative permission.

Syntax:

delete user . . .

user *user_name*
Removes user access to the device for the specified user.

Disable

Use the **disable** command to disable command completion, login from a remote console, system memory dumping, rebooting, or a specified interface.

Syntax:

disable command-completion
console-login
dump-memory . . .
interface . . .
reboot-system . . .

command-completion

Use the **disable command-completion** command to disable the automatic command completion function. See “Command Completion” on page 16 for a discussion of the automatic command completion function.

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

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

dump-memory
Disables the dumping of system memory to the installed hard disk when a serious error occurs.

reboot-system
Disables the rebooting of the system when a serious error occurs. This may

be desirable if the network service personnel wish to troubleshoot the error on-line. System rebooting cannot be disabled unless memory dumping is also disabled. If you attempt to disable system rebooting while memory dumping is enabled, system rebooting is aborted and the following message is displayed:

```
System reboot not disabled:  memory dumping must be disabled first
```

Enable

Use the **enable** command to enable command completion, login from a remote console, system memory dumping, rebooting, or a specified interface.

Syntax:

enable
- command-completion
- console-login
- dump-memory . . .
- interface . . .
- reboot-system . . .

command-completion

Use the **enable command-completion** command to enable the automatic command completion function, which assists with the command syntax. See “Command Completion” on page 16 for a discussion of the automatic command completion function.

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.
```

Attention: Before enabling console login, save the configuration with console login disabled. If login authentication is set to a remote server using Radius or Tacacs+ and the device is unable to reach the authentication server, then access to the device is denied. By disabling the console login, a lock-out situation is prevented.

dump-memory

Enables the dumping of system memory to the target device specified by the **set dump target** command (described on page on page 68) if a serious error occurs. This may be desirable so that the state of the unit at the time of the error can be preserved for troubleshooting later. The dump memory function cannot be enabled unless system rebooting is enabled. If you attempt to enable the dump memory function while system rebooting is disabled, the dump memory function is not enabled and the following message is displayed:

```
System memory dump function not enabled:  rebooting must be enabled first
```

See the **set dump enable-mode** and **set dump save-mode** commands.

Example:

```
Config> enable dump

Current System Dump Status:
   System dump is currently disabled.
   Number of existing dump files:  0

Enable system memory dumping? [No]: Yes

Current System Dump Status:
   System dump is currently enabled.
   Number of existing dump files:  0
```

Note: If you enter this command and a hard drive is not available, you will receive a message indicating that the drive is unavailable.

interface *interface#*
Causes the interface to be enabled after issuing the **reload** command.

reboot-system
Enables the rebooting of the system when a serious error occurs.

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 12. Using the Event Logging System (ELS)" on page 97 for information about ELS.

Syntax:

event

Feature

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

Syntax:

feature [*feature#* or *feature-short-name*]

All IBM 8371 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.

To access a feature's configuration prompt, enter the **feature** command followed by the feature number or short name. Table 15 on page 63 lists available feature numbers and names.

Table 15. IBM 8371 Feature Numbers and Names

Feature Number	Feature Short Name	Accesses the following feature configuration process
6	QoS	Quality of Service
17	Self Learning IP	
18	RMON	

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.

List

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

Syntax:

list configuration
devices
named-profile
patches . . .
users . . .

configuration
Displays configuration information about the device.

Example: `list configuration`

```
Hostname: [none]
Maximum packet size: [autoconfigured]
Maximum number of global buffers: [autoconfigured]
Number of spare interfaces: 0
Console inactivity timer (minutes): 0
Physical console login: disabled
Command Completion: enabled
Contact person for this node: [none]
Location of this node: [none]

Configurable Protocols:
Num Name  Protocol
11  SNMP   Simple Network Management Protocol
23  ASRT   Adaptive Source Routing Transparent Enhanced Bridge
24  HST    TCP/IP Host Services
29  MPOA   Multi-Protocol Over ATM

Configurable Features:
Num Name  Feature
6   QOS    Quality of Service
17  Self   Self Learning IP
18  RMON   Remote Network Monitor

119168 bytes of configuration memory free
```

devices [*device* **or** *devicerange***]**
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     1-port 10/100 Ethernet            Slot: 0   Port: 1
Ifc 1     1-port 10/100 Ethernet            Slot: 0   Port: 2
Ifc 2     1-port 10/100 Ethernet            Slot: 0   Port: 3
Ifc 3     1-port 10/100 Ethernet            Slot: 0   Port: 4
Ifc 4     1-port 10/100 Ethernet            Slot: 0   Port: 5
```

```
Ifc 5     1-port 10/100 Ethernet            Slot: 0   Port: 6
Ifc 6     1-port 10/100 Ethernet            Slot: 0   Port: 7
Ifc 7     1-port 10/100 Ethernet            Slot: 0   Port: 8
Ifc 8     1-port 10/100 Ethernet            Slot: 0   Port: 9
Ifc 9     1-port 10/100 Ethernet            Slot: 0   Port: 10
Ifc 10    1-port 10/100 Ethernet            Slot: 0   Port: 11
Ifc 11    1-port 10/100 Ethernet            Slot: 0   Port: 12
Ifc 12    1-port 10/100 Ethernet            Slot: 0   Port: 13
Ifc 13    1-port 10/100 Ethernet            Slot: 0   Port: 14
Ifc 14    1-port 10/100 Ethernet            Slot: 0   Port: 15
Ifc 15    1-port 10/100 Ethernet            Slot: 0   Port: 16
Ifc 16    1-port 10/100 Ethernet            Slot: 1   Port: 1
Ifc 17    1-port 10/100 Ethernet            Slot: 1   Port: 2
Ifc 18    1-port 10/100 Ethernet            Slot: 1   Port: 3
Ifc 19    1-port 10/100 Ethernet            Slot: 1   Port: 4
Ifc 20    1-port 10/100 Ethernet            Slot: 1   Port: 5
Ifc 21    1-port 10/100 Ethernet            Slot: 1   Port: 6
Ifc 22    1-port 10/100 Ethernet            Slot: 1   Port: 7
Ifc 23    1-port 10/100 Ethernet            Slot: 1   Port: 8
Ifc 24    1-port 10/100 Ethernet            Slot: 2   Port: 1
Ifc 25    1-port 10/100 Ethernet            Slot: 2   Port: 2
Ifc 26    1-port 10/100 Ethernet            Slot: 2   Port: 3
Ifc 27    1-port 10/100 Ethernet            Slot: 2   Port: 4
Ifc 28    1-port 10/100 Ethernet            Slot: 2   Port: 5
Ifc 29    1-port 10/100 Ethernet            Slot: 2   Port: 6
Ifc 30    1-port 10/100 Ethernet            Slot: 2   Port: 7
Ifc 31    1-port 10/100 Ethernet            Slot: 2   Port: 8
Ifc 32    Link Aggregation                  Slot: 3   Port: 1
Ifc 33    Link Aggregation                  Slot: 3   Port: 2
Ifc 34    Link Aggregation                  Slot: 3   Port: 3
Ifc 35    Link Aggregation                  Slot: 3   Port: 4
Ifc 36    ATM                               Slot: 1   Port: 1
Ifc 37    ATM                               Slot: 1   Port: 2
Ifc 38    ATM                               Slot: 2   Port: 1
Ifc 39    ATM                               Slot: 2   Port: 2

Ifc 40    ATM Ethernet LAN Emulation
Ifc 41    ATM Ethernet LAN Emulation
Ifc 42    ATM Ethernet LAN Emulation
Ifc 43    ATM Ethernet LAN Emulation
Ifc 44    ATM Ethernet LAN Emulation
Ifc 45    ATM Ethernet LAN Emulation
Ifc 46    ATM Ethernet LAN Emulation
Ifc 47    ATM Ethernet LAN Emulation
Ifc 48    ATM Ethernet LAN Emulation
Ifc 49    ATM Ethernet LAN Emulation
Ifc 50    ATM Ethernet LAN Emulation
Ifc 51    ATM Ethernet LAN Emulation
Ifc 52    ATM Ethernet LAN Emulation
Ifc 53    ATM Ethernet LAN Emulation
Ifc 54    ATM Ethernet LAN Emulation
Ifc 55    ATM Ethernet LAN Emulation
Ifc 56    ATM Ethernet LAN Emulation
Ifc 57    ATM Ethernet LAN Emulation
Ifc 58    ATM Ethernet LAN Emulation
Ifc 59    ATM Ethernet LAN Emulation
Ifc 60    ATM Ethernet LAN Emulation
Ifc 61    ATM Ethernet LAN Emulation
Ifc 62    ATM Ethernet LAN Emulation
Ifc 63    ATM Ethernet LAN Emulation
Config>
```

patches

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

Example:

```
list patches
Patched variable            Value

mosheap-lowmark             20
```

users Displays the users configured to access the system.

Example:

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

vpd Displays the hardware and software vital product data.

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.

Syntax:

network *interface#*

Notes:

1. If you change a user-configurable parameter, you may use the GWCON **reset interface** command, or you may **reload** the device for the change to take effect. To do so, enter the **reload** command at the OPCON prompt (*).
2. 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 device'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 device. 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 bgp-subnets

mosheap-lowmark

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.

mosheap-lowmark *new value*

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 device 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

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.

Performance

Use the **performance** command at the Config> prompt to enter the configuration environment for performance. See “Chapter 14. Configuring and Monitoring Performance” on page 155 for more information.

performance

Protocol

Use the **protocol** command at the Config> prompt to enter the configuration environment for the protocol software installed in the device.

Syntax:

protocol [*prot#* or *prot_name*]

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. 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 may be able to use the protocol’s GWCON **reset** command, or you may have to restart the device for the change to take effect. To do so, enter the **reload** 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 device.

Qconfig

Use the **qconfig** command to initiate Quick Config. Quick Config allows you to configure parameters for bridging and routing protocols without entering separate configuration environments.

Syntax:

qconfig

Set

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

Syntax:

set contact-person . . .

data-link . . .

down-notify . . .

dump enable-mode

dump save-mode

dump target

global-buffers

hostname

inactivity-timer

input-low-water

location . . .

packet-size

prompt

receive-buffers

spare-interfaces

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 device operation. It is useful for SNMP management identification of the system.

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 device to ignore transient connection problems, and smaller values will cause the device 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.

dump target

Specifies the location where the system memory image information will be written. Valid targets are the local hard disk, if one is present, or a remote host on a LAN.

If the target is a network, then IP and TFTP parameters of both the local LAN interface and the remote host are required. An additional parameter determines whether the file will be sent by TFTP as compressed or uncompressed data.

When the system dump file is sent by TFTP to the remote host, it will be written as multiple files, which must first be concatenated. For example, if the remote file was specified as /tmp/dump_to_host, and remote files are sent as compressed. The files written on the remote workstation are:

- dump_to_host0.cmp
- dump_to_host0.cm1

Depending on the total size of the dump, there may be additional files, named as:

- dump_to_host0.cm2
- dump_to_host0.cm3, and so forth.

In order to decompress and view the dump information, the files must be combined as follows into a single file (note that order is critical):

/tmp> **cat dump_to_host0.cmp dump_to_host0.cm1 dump_to_host0.cm2 dump_to_host0.cm3 > dump_to_host0_cat.cmp**

As a result, the combined file dump_to_host0_cat.cmp will contain a complete system memory dump image.

If the file was sent by TFTP as uncompressed, the file extensions are .unc, .un1, .un2, and .un3 instead of .cmp, .cm1, .cm2, and .cm3. The uncompressed files must also be concatenated to create a complete system memory dump image. For Example:

/tmp>**cat dump_to_host0.unc dump_to_host0.un1 dump_to_host0.un2 dump_to_host0.un3 > dump_to_host0_cat**

Note: The output file, dump_to_host0_cat. does not require a file extension because the file is not compressed.

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 10000). To restore the default, set the value to 0. To display the setting for global-buffers, use the **list configuration** command.

hostname *name*

Adds or changes the device name. The device name is for identification only; it does not affect any device addresses. The *name* must be less than 78 characters and is case sensitive.

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.

input-low-water *interface# low_ #_of_receive_buffers*

Allows you to configure an interface's low threshold for receive buffers. When the current number of receive buffers for an interface is less than the interface's low threshold, the packet is eligible for flow control (dropping) if the packet is queued on an output queue that has reached its high threshold (fair) value. See the description of the GWCON **queue** command for more details on flow control.

Lowering the low threshold 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 underruns occur because the receive buffer queue is empty. Raising the value has the opposite effect. To determine if underruns are occurring, use the GWCON **interface** command and specify the interface number. To determine if packets from this interface are being dropped because the low threshold has been reached, use the GWCON (Talk 5) **error** command and look at the Input Flow Drop counter value for the interface.

The range of values is 1 to 255. The default is both product-specific and device-specific. The low threshold should be less than the requested number of receive buffers. Specifying a value of 0 restores the autoconfigured default.

Use the GWCON (Talk 5) **buffer** and **queue** commands to show the low threshold setting.

Interface# is the number of the interface you are configuring.
Low_#_of_receive_buffers is the low threshold value.

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 device operation. It is useful for SNMP management identification of the system.

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.

prompt *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 16 on page 70.

Example:

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

Table 16. 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-name Feature xxxx Vx.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 to increase the receive performance of an interface and to reduce flow control drops when the router is forwarding many packets from a fast interface to a slow interface. The range of values is 5 to 1000. To restore the default, specify a value of 0. Not all device types allow the maximum number of receive buffers to be configured or support up to 1000 receive buffers. Use Table 17 to determine the default and maximum values for each device type. This command does not enforce the maximum values shown in Table 17. It allows you to configure a maximum value that is not supported by a device. The effect of this command is shown by the GWCON **buffer** command. If you configure a valid maximum value, this value appears in the Input Req column of the GWCON buffer command output. If you configure a maximum value that is not supported by the device, the GWCON **buffer** command shows the default number of receive buffers in the Input Req column and a GW subsystem ELS message is logged.

Table 17. Default and Maximum Settings for Interfaces

Interface	Default	Maximum
ATM	80	1000
10/100 Mbps Ethernet	64	1000

Time

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

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.

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**.

Valid values: -720 to 720

Default value: 0

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 device will poll the time host for the current time.

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 65 for details.

Syntax:

unpatch *variable_name*

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

Chapter 8. Using BOOT Config to Perform Change Management

This chapter describes how to use the Boot/Dump Configuration process. This chapter includes the following sections:

- "Understanding Change Management"
- "Using the Trivial File Transfer Protocol (TFTP)"

Understanding Change Management

Change management is the handling of software and configuration data for an IBM 8371. This involves:

1. Moving code and configuration data to and from the IBM 8371
2. Moving code and configuration data on the IBM 8371 system FLASH.
3. Selecting and activating specific combinations of software and configuration.

The change management functions are available by entering the **boot** command at the `Boot config>` prompt (talk 6), or the firmware should the box be in a condition where the hard drive or compact flash does not contain viable software (that is, you cannot access talk 6).

The IBM 8371 code and configuration data storage resource is divided into areas called "system banks" (banks for short), each containing a single version of the operational code and any other files pertinent to that release of the code. Up to two configuration files are associated with each bank's software.

The general change management model of the IBM 8371 is to introduce new code and/or configuration data to the system while the system runs at its present level and then activate the changed code or configuration data set later. If for some reason the new code or configuration does not function as expected, you have the ability to revert to the previous version of the configuration.

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 an IBM 8371's non-volatile configuration memory, image bank, and remote hosts.

TFTP allows you to:

- Get a configuration file from a server to an IBM 8371
- Put a configuration file from an IBM 8371 to a server

TFTP transfers involve a *client* node and a *server* node. The client node generates a TFTP Get or Put request onto the network. The IBM 8371 acts as a client node by generating TFTP requests from the IBM 8371 console using the `Boot config>` process **tftp** command.

The client can transfer a copy of a configuration file or image file stored in the image bank of a server.

© Copyright IBM Corp. 1996, 1999

Using BOOT Config

The server is any device (for example, a personal computer or workstation) that receives and services the TFTP requests. Use the ELS subsystem TFTP message log to view the transfer in progress.

Chapter 9. Configuring Change Management

This chapter describes the Change management configuration commands. It includes the following sections:

- "Accessing the Change Management Configuration Environment"
- "Change Management Configuration Commands"

Accessing the Change Management Configuration Environment

To enter the change management configuration command environment, use the CONFIG **boot** command. When the device'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**.

To return to the CONFIG process, type **exit**.

Change Management Configuration Commands

This section describes the Change Management Configuration commands. Each command includes a description, syntax requirements, and an example. Table 18 summarizes the Change Management Configuration commands.

After accessing the Change Management Configuration environment, enter the configuration commands at the `Boot config>` prompt.

Table 18. Change Management Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds an optional description to a configuration file.
Copy	Copies boot files and configuration files to or from banks.
Describe	Displays information about the stored loadfile images.
Erase	Erases a stored image or a configuration file.
List	Displays information about configuration files and scheduled load information.
Lock	Prevents the device from overwriting the selected configuration with any other configuration.
Set	Selects code bank and configuration to be used.
Tftp	Initiates TFTP file transfers between the IBM 8371 and remote servers.
Unlock	Removes the lock from a configuration allowing the configuration to be updated by the device.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

Add

Use the **add** command to add an optional description to a configuration file.

Syntax:

© Copyright IBM Corp. 1996, 1999

add

Example: Boot config> **add**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 01:26 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:30 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked

Select the source bank:  (A, B): [A]
Select the source configuration:  (1, 2): [1] 1
Enter the description of the file: () New config for
today

Attempting to set description for bank A configuration 1.
Operation completed successfully.


Boot config>list
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for today          | 01 Jan 1970 00:58 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked
```

Copy

Use the **copy** command to copy configuration files and load images to and from the banks.

Syntax:

copy	configuration *file*
	load *image*

Example:

Example: Boot config>**copy**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - CORRUPT        |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 01:26 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       |                                | 01 Jan 1970 00:14 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked


Select the source bank:  (A, B): [A]
Select the source configuration:  (1, 2): [1]
Select the destination bank:  (A, B): [B]

Select the destination configuration:  (1, 2): [1]
```

```
Copy SW configuration from: bank A, configuration 1
                        to: bank B, configuration 1.

Operation completed successfully.
```

If the copy fails you may receive one of the following messages:

Error: Active bank cannot be overwritten or erased.
You attempted to copy a configuration into the bank currently in use by the IBM 8371.

Error: File copy failed.
This condition occurs when the copy operation fails for reasons other than copying to the active configuration. The most common cause is specifying the same source and destination configurations. When you list (see "List" on page 78) the configurations, CORRUPT appears next to the bank that is damaged.

Describe

Use the **describe** command to display information about a stored image.

Syntax: describe

Example: Boot config>**describe**

```
+-------------------------+-------------------------+
|         BANK A          |         BANK B          |
| Product ID -   8371     | Product ID -   8371     |
| Version   4 Release   0 | Version   4 Release   0 |
| Mod       0 PTF       0 | Mod       0 PTF       0 |
| Feat.  2822 RPQ       0 | Feat.  2822 RPQ       0 |
| Date        31 Dec 1996 | Date        31 Dec 1996 |
+-------------------------+-------------------------+
```

Erase

Use the **erase** command to erase a configuration file.

Syntax:

erase configuration [file]

load [image]

config or load
Erases a configuration file or a load image. Enter the config number to be erased after the **erase** command.

Example: Boot config>**erase configuration**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:26 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked


Select the source bank:  (A, B): [A] A
Select the configuration to erase:  (1, 2,): [1]2
Erase SW configuration file from bank A, configuration 2.
```

```
Operation completed successfully.


Boot config>list
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                 |                   |
| CONFIG 1 - AVAIL       | test config for pubs            | 01 Jan 1970 00:14 |
| CONFIG 2 - NONE        | test config for pubs            | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                 | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for pubs            | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                 | 01 Jan 1970 00:01 |
+------------------------+---------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked
```

Notice that the list command displays **NONE** by bank A, config 2.

If the erasure fails, a message indicating the failure appears on the console with the banks that failed.

List

Use the **list** command to display information about which load images and configuration files are available and active. This command may also be used to display boot options and scheduled load information.

Syntax:

list

Example: Boot config>**list**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - AVAIL          |                                 | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs            | 01 Jan 1970 01:26 |
| CONFIG 2 - AVAIL   *   | test config for pubs            | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                 | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs            | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                 | 01 Jan 1970 00:01 |
+------------------------+---------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked

Boot config>
```

The possible file status descriptors are:

ACTIVE
The file is currently loaded and is running on the 8371

AVAIL This is a valid file that can be made ACTIVE.

CORRUPT
The file was damaged or not loaded into the 8371 completely. The file must be replaced.

LOCAL
The file will be used only on the next reload or reset. After the file is used, it will be placed in AVAIL state.

PENDING
This file will be loaded on the next reload, reset, or power-up of the 8371.

Lock

Use the **lock** command to prevent the device from overwriting the selected configuration with any other configuration.

Syntax:

lock

Example: Boot config>**lock**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970 01:03 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:26 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked

Select the source bank:  (A, B): [A]

Select the source configuration:  (1, 2): [1] 2
Attempting to lock bank A and configuration 2.

Operation completed successfully.


Boot config>list
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:13 |
| CONFIG 2 - AVAIL   L   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                |                   |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked
```

Note: Note that bank A config 2 is marked with an “L.”

Set

Use the **set** command to select the code bank and the configuration to use.

Syntax:

set

Example: Boot config>**set**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970 01:03 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:13 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked


Select the source bank:  (A, B): [A] b
Select the source configuration:  (1, 2, 3, 4): [1] 2
Select the duration to use for booting:  (once, always): [always]
Set SW to boot using bank B and configuration 2, always.
```

```
Operation completed successfully.


Boot config>list
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:13 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - ACTIVE      |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked
```

TFTP

Use the **tftp** command to initiate TFTP file transfers between the 8371 and remote servers.

Syntax:

tftp get	config
	load
tftp put	config
	load

Example: Boot config>**tftp get load**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970 01:03 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:01 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked


Specify the server IP address (dotted decimal): : [1.2.3.4] 192.9.200.1
Specify the remote file name: : (/u/bin) /usr/8371load/8371.img
Select the destination bank:  (A, B, F): [A] a
  TFTP SW load image
  get:   /usr/8371load/8371.img
  from:  192.9.200.1
  to:    bank A.

Operation completed successfully.
```

Notes:
When putting files to a server:

1. Make sure that the files on the target server have the appropriate permissions that would allow anyone to write to those files. If not, the put operation will fail.
2. You must be aware of the files you are putting to the target server.

Unlock

Use the **unlock** command to allow the device to overwrite the selected configuration that was previously locked.

Syntax:

unlock

Example: Boot config>**unlock**

```
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                | 01 Jan 1970 01:03 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:13 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970 00:01 |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL   L   |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked


Select the source bank:  (A, B): [A] B
Select the source configuration:  (1, 2): [1] 2
Attempting to unlock bank B and configuration 2.

Operation completed successfully.
Boot config>list
+------ BankA -----------+--------- Description ----------+------ Date -------+
| IMAGE - NONE           |                                |                   |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:01 |
| CONFIG 2 - AVAIL   *   | test config for pubs           | 01 Jan 1970 01:13 |
+------ BankB -----------+--------- Description ----------+------ Date -------+
| IMAGE - ACTIVE         |                                | 01 Jan 1970       |
| CONFIG 1 - AVAIL       | test config for pubs           | 01 Jan 1970 00:54 |
| CONFIG 2 - AVAIL       |                                | 01 Jan 1970 00:01 |
+------------------------+--------------------------------+-------------------+
  * - Last Used Config      L - Config File is Locked
```

Note: Note that bank A config 2 is no longer marked with an “L.”

Chapter 10. The Operating/Monitoring Process (GWCON - Talk 5) 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 device user interface.

Using GWCON commands, you can:

- List the protocols and interfaces currently configured in the device.
- 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.

The GWCON command interface is made up of levels called modes. Each mode has its own prompt. For example, the prompt for the SNMP protocol is `SNMP>`.

If you want to know the process and mode you are communicating with, press **enter** 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 GWCON from OPCON (*), choose one of the following methods:

1. Enter the OPCON **console** command:

   ```
   * console
   ```

2. At the OPCON prompt, enter the **status** command to find the PID of GWCON. (See page 9 for a sample output of the **status** command.)

   ```
   * status
   ```

   Then, enter the **talk** command followed by the PID number for GWCON:

   ```
   * talk 5
   ```

The console displays the GWCON prompt (+). If the prompt does not appear, press **enter**. Now you can enter GWCON commands.

To return to OPCON, enter the OPCON intercept character. (The default is **Ctrl-P**.)

© Copyright IBM Corp. 1996, 1999

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 19.

To use the GWCON commands, access the GWCON process by entering **talk 5** and enter the GWCON commands at the (+) prompt.

Table 19. GWCON Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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.
Error	Displays error counts.
Event	Enters the Event Logging System environment.
Feature	Provides access to console commands for independent device features outside the usual protocol and network interface console processes.
Interface	Displays network hardware statistics or statistics for the specified interface.
Memory	Displays memory, buffer, and packet data.
Network	Enters the console environment of the specified network.
Performance	Provides a snapshot of the main processor utilization statistics.
Protocol	Enters the command environment of the specified protocol.
Queue	Displays buffer statistics for a specified interface.
Reset	Disables the specified interface and then re-enables it using new interface, protocol and feature configuration parameters.
Statistics	Displays statistics for a specified interface.
Test	Enables a disabled interface or tests the specified interface.
Uptime	Displays time statistics for the device.

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#* or]

Example:

Nt Network interface number associated with the software.

Interface
Type of interface.

Input Buffers:

Req Number of receive buffers requested. This is either the device's default number of receive buffers or a valid value set with the CONFIG (Talk 6) **set receive-buffers** command.

Notes:

1. If this column is 0 for an interface, then this is a virtual interface for which receive buffers are not allocated. In this case, the virtual interface uses the receive buffers of the device that it is mapped to. For example, a dial circuit interface uses the receive buffers of its base net or interface.
2. If you specify a value on the CONFIG **set receive-buffers** command that is not supported by the device, then the number of buffers requested is equal to the device's default number of receive buffers.

Alloc Number of receive buffers allocated.

Note: The number of receive buffers allocated is less than the number of receive buffers requested if there is not enough memory available to allocate the requested number of buffers.

Low The device's low threshold for receive buffers. When the current number of receive (input) buffers for an interface is less than the interface's low threshold, the packet is eligible for flow control (dropping). See the description of the GWCON (Talk 5) **queue** command for more details on flow control. The low threshold is configurable using the CONFIG (Talk 6) **set input-low-water**command.

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 device's network interfaces. This command is useful when tracking changes in large counters. Using this command does not save space or speed up the device.

Enter the interface (or net) number as part of the command. To get the interface number, use the GWCON **configuration** command.

Syntax:

clear

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 device 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:

- The first line gives the product name.
- The second line lists the program/product number, Feature Number, Version, Release, PTF and RPQ information.
- The remaining lines list the configured protocols, followed by the configured features.

The following information is displayed for protocols:

Num Number that is associated with the protocol.

Name Abbreviated name of the protocol.

Protocol
Full name of the protocol.

The following information is displayed for features:

Num Number associated with the feature.

Name Abbreviated name of the feature.

Feature
Full name of the feature.

The following information is displayed for 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 device 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 device 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 device 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 device 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.

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 **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.

Syntax:

disable interface *interface#*

Error

Use the **error** command to display error statistics for the network. This command provides a group of error counters.

Syntax:

error

Example:

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 device 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 device is reinitialized. See “Chapter 12. Using the Event Logging System (ELS)” on page 97 for information about the Event Logging System and its commands. Use the **exit** command to return to the GWCON process.

Syntax:

event

Feature

Use the **feature** command to access console commands for specific IBM 8371 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.

To access that feature's console prompt, enter the **feature** command at the GWCON prompt followed by the feature number or short name. Table 15 on page 63 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*

Interface

Use the **interface** command to display statistical information about the network interfaces (for example, Ethernet). This command can be used without a qualifier to provide a summary of all the interfaces 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`

Note: The following information may be displayed. The display varies depending on the device.

Nt Global interface number.

Interface
Interface name.

Slot-Port
Slot number and port number of the interface.

Port Name
Port number, if applicable on the slot.

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.

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 device operations. The number of free buffers should remain constant when the device is idle. If it does not, contact your service representative.

Syntax:

memory

Example:

```
memory
Physical installed memory:        16 MB
Total routing (heap) memory:      12 MB
Routing memory in use:            13 %

                  Total  Reserve    Never     Perm     Temp     Prev
                                     Alloc    Alloc    Alloc    Alloc
Heap memory    12231155    26488 10687312  1438487   104924      432

Number of global buffers: Total = 300, Free = 300, Fair = 77, Low = 60
Global buff size: Data = 2048, Hdr = 17, Wrap = 72, Trail = 65, Total = 2208
```

Physical installed memory
The total amount of physical RAM installed in the device.

Total routing memory
The amount of memory available to the routing function, not including that allocated to the base operating system, system extensions, or options such as APPN. This is also called ″heap″ memory, and matches the ″Total″ heap memory size given in bytes shortly thereafter.

Routing memory in use
The percentage of total routing memory that is currently being used by the routing function. Heap memory currently in use is counted under the following headings **Perm Alloc** and **Temp Alloc**.

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 device tasks.

Temp Alloc
Memory allocated temporarily to device 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. This command obtains the console prompt for the specified interface.

Syntax:

network *interface#*

At the GWCON prompt (+), enter the **configuration** command to see the protocols and networks for which the device is configured. See “Configuration” on page 86 for more information on the configuration command.

Enter **interface** at the + prompt for a display of the networks for which the device is configured.

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

```
+network 0
ATM+
```

In the example, the ATM+ prompt is displayed. You can then view information about the ATM interface by entering the ATM operating commands.

After identifying the interface number of the interface you want to monitor, for interface-specific information, see the corresponding 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:

- ATM
- Ethernet
- Ethernet LECs

- LAG

Performance

Use the **performance** command at the GWCON prompt to enter the monitoring environment for performance. See “Chapter 14. Configuring and Monitoring Performance” on page 155 for more information.

Protocol

Use the **protocol** command to communicate with the device software that implements the network protocols installed in your device. 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.

Syntax:

protocol *prot#*

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 “Accessing the Configuration Process, CONFIG (Talk 6)” on page 11 for instructions on accessing Config>. To return to GWCON, enter **exit**.

See the corresponding monitoring chapter in this manual or in the *Protocols and Features* for information on a specific protocol’s console commands.

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.

Syntax:

queue *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.

Nt Network interface number associated with the software.

Interface
Type of interface.

Input Queue:

Alloc Number of buffers allocated to this device.

Low The low threshold for receive (input) buffers is used to activate flow control for this device. The low threshold is configurable using the CONFIG (Talk 6) **set input-low-water** command.

Curr Current number of buffers on this device. The value will be 0 if the device is disabled.

Output Queue:

Fair The high threshold for the interface's output queue when flow control is activated for an input device.

Note: When Bandwidth Reservation (BRS) is configured for PPP and Frame Relay interfaces, the output fair value is ignored and the queue lengths configurable with BRS are used to determine if a packet should be discarded due to flow control.

Curr Number of packets currently waiting to be transmitted on this device. The eligibility discard depends on the global low water mark described in the **memory** command.

If a packet is received and the input queue current value is less than the input queue low threshold value, then the packet will be subject to flow control. For locally originated packets, a packet is subject to flow control if the number of free global buffers is less than the low threshold for global buffers. If a packet subject to flow control is to be transmitted on a device which has an output queue current value that is greater than the output queue high threshold (fair), then the packet is dropped instead of queued. When a packet is dropped due to flow control, the output discards counter is incremented and ELS event GW.036 or GW.057 is logged. If the packet was not locally originated, the input flow drop counter for the input interface is incremented. The output discards and input flow drop counters are displayed by the GWCON **error** command.

Due to the scheduling algorithms of the device, 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.

Reset

Use the **reset** command to disable the specified interface and then re-enable it using new interface, protocol and feature configuration parameters. See "Resetting Interfaces" on page 55 for more information.

Syntax:

reset *interface#*

Statistics

Use the **statistics** command to display statistical information about the network software, such as the configuration of the networks in the device.

Syntax:

statistics *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.

Example:

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.

Syntax:

test *interface#*

Note: For this command to work, you must enter the ***complete*** name of the command followed by the interface number.

Enter the interface or network number as part of the command. To obtain the interface number, use the GWCON **configuration** command. For example, when testing starts, the console displays the following message:

```
Testing net 0 Eth/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
Testing net 0 ATM/0 ...successful
Testing net 0 ATM/0 ...failed
Testing net 0 ATM/0 ...still testing
Network is already undergoing test, attempting restart
```

Some interfaces may take more than 30 seconds before testing is done.

Uptime

Use the **uptime** command to display time elapsed since the last reload.

Syntax:

uptime

Chapter 11. The Messaging (MONITR - Talk 2) Process

This chapter explains how to collect and display messages. (Refer to “Chapter 12. Using the Event Logging System (ELS)” on page 97 for information about ELS and message formats. Refer also to the *Event Logging System Messages Guide* for a description of each message. This chapter includes the following sections:

- “What is Messaging (MONITR)?”
- “Commands Affecting Messaging”
- “Entering and Exiting the Messaging (MONITR) Process”
- “Receiving Messages”

What is Messaging (MONITR)?

The MONITR process provides a view of activity inside the device and the networks. MONITR also displays logging messages from the software.

Commands Affecting Messaging

The following commands affect the messaging process:

- OPCON commands:
  - **divert** temporarily diverts output to a different device.
  - **flush** causes the software to discard the messages it collects.
  - **halt** reverses the action of the divert command.
  - **talk** displays message output.
- CONFIG **set logging disposition** command sets the initial device to which the software sends its output.

Entering and Exiting the Messaging (MONITR) Process

To enter the messaging process from OPCON enter the **event** command or the **talk 2** command.

The console displays the messages the software has accumulated.

To exit messaging and return to OPCON, enter the OPCON intercept character (the default is **Ctrl-P**).

Receiving Messages

To receive messages at your console, enter the messaging process as described in the previous section. The software then displays all the messages it has recorded since it was last invoked. While you are connected to the messaging process, it displays all messages as they arrive.

Use the OPCON **divert** and **halt** commands to view software messages while you are doing something else with the device. Permitted devices divert output to TTY0 (the local console), TTY1, or TTY2 (the remote consoles).

© Copyright IBM Corp. 1996, 1999

Chapter 12. Using the Event Logging System (ELS)

This chapter describes the Event Logging System (ELS). The ELS continually logs all events, filtering them according to parameters that you select. A combination of operational 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 device operating system. ELS manages the messages logged as a result of device activity. Use ELS commands to set up a configuration that sorts out only those messages you feel are important. You can then display the messages on the console terminal screen, log them to a remote workstation, or send the messages to a network management station using Simple Network Management Protocol (SNMP) traps.

The ELS system and the operational counters are the best troubleshooting tools you have to isolate problems in the device. A quick scan of the event messages will tell you whether the device has a problem and 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 the device reinitializes.

Occasionally, it is helpful to temporarily view messages using parameters other than was set up in the ELS configuration environment, without having to reinitialize the device. The ELS operating and monitoring environment is used to:

- Temporarily change the default ELS display settings
  - Changes made in the ELS console environment take effect immediately
  - Changes made in the operating/monitoring environment are not stored in nonvolatile configuration storage.
- View statistical information regarding ELS uses of dynamic RAM

Note: Specific ELS messages are described in the *Event Logging System Messages Guide*.

ELS is a subprocess that you access from the OPCON process.

© Copyright IBM Corp. 1996, 1999

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 device. 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 device does not have a default ELS configuration. You must enter the ELS configuration environment and set the default state.

To enter the ELS configuration environment from OPCON:

1. Enter the **configuration** command. The console displays the CONFIG prompt (`Config>`). If the prompt does not appear when you first enter CONFIG, press **enter**.
2. 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.

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 accept the subsystem, event number, and logging level as parameters.

Causes of Events

Events occur continuously while the device 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 10 shows the message contents.

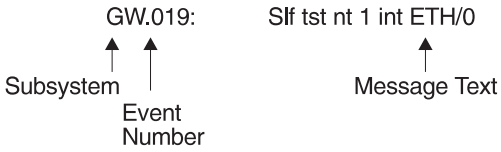

Figure 10. Message Generated by an Event

The information illustrated in Figure 10 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 device component, such as a protocol or interface. In Figure 10, **GW** identifies the subsystem through which this event occurred.

Other examples of subsystems include IP and ETH. On a particular device, the actual subsystems present depend on the hardware and software configured for that device. You can use the **list subsystem** command described in this chapter to see a list of the subsystems on your device.

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 10, **019** 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 identifier, 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. Use the **list subsystem** ELS console command to display the setting of the logging level. Table 20 on page 100 lists the logging levels and types. ERROR, INFO, TRACE, STANDARD, and ALL are aggregates of other logging level types. STANDARD is the recommended default.

Table 20. 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

The logging level setting affects the operation of the following commands:

- **Display subsystem**
- **Nodisplay subsystem**
- **Trap subsystem**
- **Notrap subsystem**
- **Remote subsystem**
- **Noremote 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 ETH 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 10 on page 99, `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 21 describes the error or packet completion codes. Packet completion codes indicate the disposition of the packets received by the device.

Table 21. 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

Table 21. Packet Completion Codes (Error Codes) (continued)

Code	Meaning
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 device 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, 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, 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, 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.

Using ELS

To use ELS effectively, do the following:

- Know what you want 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 *Event Logging System Messages Guide* to determine which messages are not normal.

Using ELS

When initially viewing ELS from the MONITR process, you will see a considerable amount of information. Because the device 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
```

Managing ELS Message Rotation

It is also important to note that the ELS messages continually rotate through the device'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 device. You can also do this by attaching a PC to the device'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

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, set up ELS for the messages you want to capture using the ELS console commands in "Chapter 13. Configuring and Monitoring the Event Logging System (ELS)" on page 113.

To capture the ELS output to a file on an AIX or UNIX host, follow these steps:

1. From the host, enter **telnet** *device_ip_addr* **| tee** *local_file_name*
   - *device_ip_addr* is the IP address of the device
   - *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.

You can also use remote logging instead of capturing ELS output on a UNIX Host. For more information about remote logging, see "Using and Configuring ELS Remote Logging" on page 104.

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 IBM

8371. When ELS is configured appropriately, an SNMP trap will be generated each time the selected event occurs. For more information about SNMP, see *Protocols and Features*.

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, type:

```
trap event eth.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 need to reboot to activate these changes.
2. Enable your network management station to receive and properly display the enterprise-specific traps.

Follow these steps to trap groups, subsystems, and events.

Using ELS to Troubleshoot a Problem

If you are trying to troubleshoot a particular problem, display the messages related to the problem. For example, if experiencing a problem with bridging, turn on the bridging messages:

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 them up in the *Event Logging System Messages Guide* 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

Router cannot communicate with an IPX server on an Ethernet.

1. 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**.
2. At the GWCON prompt (+), enter **IPX** to access the IPX console prompt (IPX>).
3. At the IPX console prompt, enter the **slist** command to verify that the server is listed. (See the section on monitoring IPX in the *Protocols and Features* for information on the **slist** command.)
4. Check the IPX configuration.
5. 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.001. This indicates that the server has a bad Ethernet type field.

Using and Configuring ELS Remote Logging

The remotely-logged ELS message contains all of the information that is contained in ELS messages found in the monitor queue, as viewed under `talk 2`, and also contains additional information as shown in Figure 11.

```
Date/Time         IP address      Sequence Number      Local Name      ELS Subsystem Name, &
                  assigned        used for detecting   assigned        Formatted message
                  by the user     missing messages     by the user

Nov 20 12:13:47   5.1.1.1         Msg [0444] from      ** IBM/8371 **  :els: MPC.011 Del ent ...
```

Figure 11. Syslog Message Description

Note the following differences in the remote log display:

- The month and day of month in addition to the time, which is always displayed as the time-of-day.
- An IP address, which is the user-specified source IP address. If a DNS server resolves the source IP address to a hostname, then the hostname will be displayed instead of the IP address.
- A Sequence number is added to the message by the source device to assist in detecting dropped messages. See “Remote Logging Output” on page 108 for an explanation of dropped messages. When the sequence number of the message reaches 9999, the next sequence number is 0001.
- A “Local Name” for the source device, to assist in distinguishing between messages from multiple sources. If you do not configure a local name, this field is blank.

Syslog Facility and Level

Remotely-logged ELS messages are transmitted over the network in UDP packets with the destination port number in the UDP header always equal to 514, the syslog port. To receive and process the UDP packets, the *syslog daemon* (syslogd) must be running in the remote workstation that is receiving and logging the ELS messages. See “Remote Workstation Configuration” on page 105 for details.

Although it is not displayed in the remotely-logged ELS message, every ELS message sent on the network in a UDP packet must be assigned a *syslog_facility* and a *syslog_level*. The syslog daemon uses the combination of facility and level to determine where to route the message. Typically, you want the ELS messages to be written to one or more files in the remote host. Other options include displaying the message on the console, sending the message to one or more users, or sending the message to another workstation.

The commands you use to specify the *syslog_facility* and *syslog_level* values, along with other remote-logging related console commands, are described in “ELS

Monitoring Commands" on page 130 and "ELS Configuration Commands" on page 113 . Review these commands before reading through the next section.

Remote Workstation Configuration

The following configuration assumes that a single 8371 is remote-logging to a single remote workstation. You can configure multiple 8371s to remote-log to the same remote workstation. However, a particular 8371 can log to one and only one remote workstation. The operating system used in this example is AIX 4.2. Your environment may be slightly different. For more information on syslog, refer to the documentation for your operating system.

To perform the configuration on an AIX workstation, you must log in as **root**. To configure the workstation:

1. Create or edit a syslog.conf file to specify where ELS messages with particular *syslog_facility* and *syslog_level* values are to be written. See the bottom of Figure 12 on page 106 for an example of how to specify the message destination. Note that the full pathname of the log files must be specified. The default location for the syslog configuration file is /etc/syslog.conf.
2. Create the files for logging syslog messages that you specified in the syslog.conf file.
3. Start the syslog daemon by entering **syslogd**. To start the syslog daemon from SRC (System Resource Controller), enter **startsrc -s syslogd**. If the pathname of the configuration file is not /etc/syslog.conf, then enter **syslogd -f** *pathname*. To start the syslog daemon in debug mode, enter **syslogd -d**.

   Note: Running multiple instances of the syslog daemon is not supported.
4. If the syslog daemon is already running when you create or modify the syslog.conf file, it must be restarted so that the daemon reinitializes the configuration from syslog.conf.
5. Verify the setup by using the **logger** command as follows:

   ```
   logger -p user.alert  THIS IS A TEST MESSAGE (user.alert)
   logger -p news.info  THIS IS A TEST MESSAGE (news.info)
   ```

   If the setup is correct, `THIS IS A TEST MESSAGE...` will be written to the files specified in syslog.conf.

```
# @(#)34       1.9 src/bos/etc/syslog/syslog.conf, cmdnet, bos411, 9428A410j 6/13/93 14:52:39
#
# COMPONENT_NAME: (CMDNET) Network commands.
#
# FUNCTIONS:
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1988, 1989
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
# /etc/syslog.conf - control output of syslogd
#
# Each line must consist of two parts:-
#
# 1) A selector to determine the message priorities to which the
#    line applies
# 2) An action.
#
# The two fields must be separated by one or more tabs or spaces.
#
# format:
#
# <msg_src_list>               <destination>
#
# where <msg_src_list> is a semicolon separated list of <facility>.<priority>
# where:
#
# <facility> is:
#      * - all (except mark)
#      kern,user,mail,daemon, auth, syslog, lpr, news, uucp, cron, authpriv, local0 - local7
#
# <priority or level> is one of (from high to low):
#      emerg,alert,crit,err(or),warn(ing),notice,info,debug
#      (meaning all messages of this priority or higher)
#
# <destination> is:
#      /filename - log to this file
#      username[,username2...] - write to user(s)
#      @hostname - send to syslogd on this machine
#      * - send to all logged in users
#
# example:
# "mail messages, at debug or higher, go to Log file. File must exist."
# "all facilities, at debug and higher, go to console"
# "all facilities, at crit or higher, go to all users"
#  mail.debug          /usr/spool/mqueue/syslog
#  *.debug             /dev/console
#  *.crit                      *

#   syslog messages with facilty / priority values of LOG_USER,   LOG_ALERT
user.alert            /tmp/syslog_user_alert

#   syslog messages with facilty / priority values of LOG_NEWS,  LOG_INFO
news.info           /tmp/syslog_news_info
```

Figure 12. syslog.conf Configuration File

Configuring the 8371 for Remote Logging

To configure a 8371:

1. In talk 6, configure the remote-logging facility as shown in Figure 13 on page 107. The IP address specified as the *source-ip-addr* should be an IP address that is configured in the 8371 for easier identification when the IP address or the hostname is shown in the remotely-logged ELS message. You should also verify that this IP address resolves quickly into a hostname by the

name server or that the name server at least responds quickly with “address not found.” To determine whether this happens, issue the **host** command on your workstation as follows:

```
workstation> host 5.1.1.1
host: address 5.1.1.1 NOT FOUND
workstation>
```

If the response takes more than 1 second, select an IP address which resolves more quickly.

2. In talk 6 configure events and subsystems for remote-logging, as shown in Figure 14 on page 108.
3. Write the configuration and reload the device.

```
ELS config>set remote source-ip-addr 5.1.1.1
Source IP Addr = 5.1.1.1

ELS config>set remote remote-ip-addr 192.9.200.1
Remote Log IP Addr = 192.9.200.1

ELS config>set remote local-id ** IBM/8371 **
Remote Log Local ID = ** IBM/8371 **

ELS config>set remote no-msgs-in-buffer 100
Number of messages in Remote Log Buffer must be 100-512
Number of Messages in Remote Buffer = 100

ELS config><B>set remote facility log_news
Default Syslog Facility = LOG_NEWS

ELS config>set remote level log_info
Default Syslog Level = LOG_INFO

ELS config>set remote on
Remote Logging is ON

ELS config>list remote

------------------   Remote Log Status  -----------------

Remote Logging is ON
Source IP Address = 5.1.1.1
Remote Log IP Address = 192.9.200.1
Default Syslog Facility = LOG_NEWS
Default Syslog Priority Level = LOG_INFO
Number of Messages in Remote Log =  100
Remote Logging Local ID = ** IBM / 8371 **
ELS config>
```

Figure 13. Configuring the 8371 for Remote Logging

```
ELS config>display sub snmp all
ELS config>remote sub snmp all log_news log_info

ELS config>display event srt.017
ELS config>remote event srt.017 log_news log_info

ELS config>display event stp.016
ELS config>remote event stp.016 log_user log_info

ELS config>display event stp.026
ELS config>remote event stp.026 log_news log_info

ELS config>display event stp.024
ELS config>remote event stp.024 log_news log_info


ELS config>list status
Subsystem:    SNMP
Disp levels:  ERROR INFO TRACE
Trap levels:  none
Trace levels: none
Remote levels:  ERROR INFO TRACE
        Syslog Facility/Level: LOG_NEWS LOG_INFO

Event     Display Trap    Trace     Remote
SRT.017   On      Unset   Unset     On
                                    Syslog Facility/Level: LOG_NEWS LOG_INFO
STP.016   On      Unset   Unset     On
                                    Syslog Facility/Level: LOG_NEWS LOG_INFO
STP.026   On      Unset   Unset     On
                                    Syslog Facility/Level: LOG_NEWS LOG_INFO
STP.024   On      Unset   Unset     On
                                    Syslog Facility/Level: LOG_NEWS LOG_INFO
```

Figure 14. Configuring Subsystems and Events for Remote Logging

Remote Logging Output

Figure 15 on page 109 shows a sample from the /tmp/syslog_news_info file. Notice that the first message has a sequence number of 310. This means that the first 309 ELS messages were not sent from the source 8371. There are several reasons for this:

- The remote-logging facility had not completed initialization when the messages were first passed to ELS
- A route from the source 8371 to the remote workstation was not in the routing table
- The interface for the outbound UDP packet containing the ELS messages was not in the “Up” state

```
Nov 20 12:03:16 workstа01 root:  THIS IS A TEST MESSAGE  (news.info)
Nov 20 12:08:48 5.1.1.1 Msg [0310] from ** IBM / 8371 **: els:  IP.022: add nt 192.9.200.0 int 192.9.200.20
nt 0 int Eth/0

 1  ( messages 311, 312, and 313 did not get remote-logged due to ARP request outstanding - see
  explanation in the text)

 2  (messages 314 and 315 were logged to a separate
file - see explanation in the text)

Nov 20 12:08:48 5.1.1.1 Msg [0316] from ** IBM / 8371 **: els:  IP.068: routing cache cleared
Nov 20 12:08:48 5.1.1.1 Msg [0317] from ** IBM / 8371 **: els:  IP.022: add nt 5.0.0.0 int 5.1.1.1 nt 5 int Eth/4
Nov 20 12:08:48 5.1.1.1 Msg [0318] from ** IBM / 8371 **: els: SRT.017: Enabling SRT on port 5 nt 5 int Eth/4


(message 319 was logged to a separate file)

Nov 20 12:08:48 5.1.1.1 Msg [0320] from ** IBM / 8371 **: els:  IP.068: routing cache cleared

(120 messages not shown)

Nov 20 12:13:33 5.1.1.1 Msg [0441] from ** IBM / 8371 **: els:  GW.022: Nt fld slf tst nt 3 int Eth/3
Nov 20 12:13:33 5.1.1.1 Msg [0442] from ** IBM / 8371 **: els:  GW.022: Nt fld slf tst nt 6 int Eth/5
Nov 20 12:13:38 5.1.1.1 Msg [0443] from ** IBM / 8371 **: els:  GW.022: Nt fld slf tst nt 11 int ISDN/0


(messages 444 and 447 were logged to a separate file)


(messages 445 and 446 did not get remote-logged due to ARP request outstanding)

Nov 20 12:13:50 5.1.1.1 Msg [0448] from ** IBM / 8371 **: els:  GW.022: Nt fld slf tst nt 4 int PPP/0
Nov 20 12:13:50 5.1.1.1 Msg [0449] from ** IBM / 8371 **: els:  IP.068: routing cache cleared
Nov 20 12:13:50 5.1.1.1 Msg [0450] from ** IBM / 8371 **: els:  IP.058: del nt 4.0.0.0 rt via 0.0.0.4 nt 4 int PPP/0
```

Figure 15. Sample Contents from Syslog News Info File

If the initial ELS messages that are generated during and immediately after booting are of particular interest, then it is recommended that these messages also be displayed in the monitor queue, which is viewed with talk 2. Figure 16 on page 110 shows the talk 2 output including the initial messages that did not get remote-logged. Note that there is a message in the talk 2 output that indicates that the remote-logging facility is available. This does not indicate that a route exists to the remote workstation, nor that the associated interface is in the “Up” state. It simply provides a reference point before which no messages can be successfully remote-logged.

Also notice that you can account for the messages that were missing (indicated in Figure 15 with **2**) in the talk 2 output.

```
12:08:17 SNMP.024: generic trc (P2) at snmp_mg.c(766): Now 0 trap destinations
12:08:17 SNMP.012: comm public added
12:08:17 SNMP.012: comm public added
12:08:27 SNMP.022: ext err (Z1) at snmp_resconf.c(322): add_device_if_info(): sr
rdrec failed

12:08:27 SNMP.022: ext err (Z1) at snmp_resconf.c(322): add_device_if_info(): sr
rdrec failed

12:08:27 SNMP.028: err (E2) at snmp_moh.c(1583) : Duplicate
12:08:27 SNMP.028: err (E2) at snmp_moh.c(1583) : Duplicate
12:08:28   GW.022: Nt fld slf tst nt 13 int PPP/3
12:08:28   IP.022: add nt 4.0.0.0 int 4.1.1.1 nt 4 int PPP/0

     ( 297 messages not shown )                                    Corresponding Sequence
                                                                   Numbers in
12:08:43   GW.022: Nt fld slf tst nt 12 int PPP/2                  Remote-Logging Files :
12:08:43   GW.022: Nt fld slf tst nt 13 int PPP/3
12:08:48   IP.022: add nt 192.9.200.0 int 192.9.200.20 nt 0 int Eth/0   [0310] first message logged
12:08:48  SRT.017: Enabling SRT on port 1 nt 0 int Eth/0          -- not logged (ARP request) --
12:08:48  STP.016: Select as root TB-1, det topol chg             -- not logged (ARP request)--
12:08:48  STP.026: Root TB-1, strt hello tmr                      -- not logged (ARP request)--
12:08:48  ARP.002: Pkt in 1 1 800 nt 0 int Eth/0                  [0314]
12:08:48  ARP.002: Pkt in 2 1 800 nt 0 int Eth/0                  [0315]
12:08:48   IP.068: routing cache cleared                          [0316]


     ( 126 messages not shown )

12:13:38   GW.022: Nt fld slf tst nt 11 int ISDN/0                [0443]
12:13:47  ARP.011: Del ent 1 3 nt 0 int Eth/0                     [0444]
12:13:47  ARP.011: Del ent 1 3 nt 0 int Eth/0                     -- not logged (ARP request) --
12:13:47  ARP.002: Pkt in 1 1 800 nt 5 int Eth/4                  -- not logged (ARP request)--
12:13:47  ARP.002: Pkt in 2 1 800 nt 0 int Eth/0                  [0447]
12:13:50   GW.022: Nt fld slf tst nt 4 int PPP/0                  [0448]
```

Figure 16. Output from Talk 2

You can use the timestamp, which appears in both the remote-logging output file and the talk 2 output, to determine when the first ELS message is successfully remote-logged. To use the timestamp for this purpose, configure ELS such that the timestamp in the monitor queue displays the time-of-day.

Additional Considerations

ELS Messages Containing IP Addresses

ELS messages containing an IP address which matches the IP address of the remote workstation will not be remote-logged, even if configured for remote-logging, and may appear under talk 2. These messages are discarded instead of being remote-logged in order to prevent excessive UDP packets from being sent on the network.

Duplicate Logging

If a facility value is repeated in *syslog.conf*, for example:

```
user.debug        /tmp/syslog_user_debug
user.alert        /tmp/syslog_user_alert
```

The syslog daemon will log *user.debug* messages only to the */tmp/syslog_user_debug* file while user.alert messages will be logged to both the */tmp/syslog_user_debug* file and the */tmp/syslog_user_alert* file. This is consistent with the syslog design that logs the more severe conditions in multiple places.

To prevent this duplicate logging, it is recommended that different facility values be specified in the *syslog.conf* file. A total of 19 facility values are available.

Recurring Sequence Numbers in Syslog Output Files

Depending upon the configuration of your network, it is possible for duplicate UDP packets containing ELS messages to arrive at the remote host. It is also possible for the packets to arrive in a different order than they were transmitted. An example of this phenomenon is shown in Figure 17. Notice that the messages with sequence numbers 628 through 633 are logged twice. Also notice that after the first occurrence of sequence number 0630, sequence number 0629 occurs again, followed by the second occurrence of 0630.

```
Apr 01 10:48:33 0.0.0.0 Msg [0628] from: RA22: : els: IPX.018: SAP gen rply sent nt 5 int TKR/1, 1 pkts
Apr 01 10:48:33 0.0.0.0 Msg [0628] from: RA22: : els: IPX.018: SAP gen rply sent nt 5 int TKR/1, 1 pkts
Apr 01 10:49:08 0.0.0.0 Msg [0629] from: RA22: : els: IPX.037: RIP resp sent nt 0 int TKR/0, 1 pkts
Apr 01 10:49:08 0.0.0.0 Msg [0630] from: RA22: : els: IPX.018: SAP gen rply sent nt 0 int TKR/0, 1 pkts
Apr 01 10:49:08 0.0.0.0 Msg [0629] from: RA22: : els: IPX.037: RIP resp sent nt 0 int TKR/0, 1 pkts
Apr 01 10:49:08 0.0.0.0 Msg [0630] from: RA22: : els: IPX.018: SAP gen rply sent nt 0 int TKR/0, 1 pkts
Apr 01 10:49:33 0.0.0.0 Msg [0631] from: RA22: : els: IPX.037: RIP resp sent nt 5 int TKR/1, 1 pkts
Apr 01 10:49:33 0.0.0.0 Msg [0631] from: RA22: : els: IPX.037: RIP resp sent nt 5 int TKR/1, 1 pkts
Apr 01 10:49:33 0.0.0.0 Msg [0632] from: RA22: : els: IPX.018: SAP gen rply sent nt 5 int TKR/1, 1 pkts
Apr 01 10:49:33 0.0.0.0 Msg [0632] from: RA22: : els: IPX.018: SAP gen rply sent nt 5 int TKR/1, 1 pkts
Apr 01 10:50:08 0.0.0.0 Msg [0633] from: RA22: : els: IPX.037: RIP resp sent nt 0 int TKR/0, 1 pkts
Apr 01 10:50:08 0.0.0.0 Msg [0633] from: RA22: : els: IPX.037: RIP resp sent nt 0 int TKR/0, 1 pkts
```

Figure 17. Example of Recurring Sequence Numbers in Syslog Output

Because neither Syslog nor UDP has the ability to handle duplicate or out of sequence packets, it is important to recognize the possibility of duplicate sequence numbers occurring.

Chapter 13. Configuring and Monitoring the Event Logging System (ELS)

This chapter describes how to configure events logged by ELS and how to use the ELS commands. The information includes the following sections:

- "Accessing the ELS Configuration Environment"
- "ELS Configuration Commands"
- "Entering and Exiting the ELS Operating Environment" on page 130
- "ELS Monitoring Commands" on page 130

For more information on the Event Logging System and how to interpret ELS event messages, refer to "Chapter 12. Using the Event Logging System (ELS)" on page 97.

Accessing the ELS Configuration Environment

The ELS configuration environment is characterized by the `ELS config>` prompt. Commands entered at this prompt are described "Chapter 13. Configuring and Monitoring the Event Logging System (ELS)".

To enter the ELS configuration environment:

1. Enter **configuration**.

   The monitoring displays the `Config>` prompt. If the prompt does not appear, press **enter**.
2. At the `Config>` prompt, enter the following command to access ELS:

   ```
   event
   ```

   The monitoring displays the ELS configuration prompt (`ELS config>`). Now, you can enter ELS configuration commands.

To leave the ELS configuration environment, enter the **exit** command.

ELS Configuration Commands

Table 22 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 22. ELS Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
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.

© Copyright IBM Corp. 1996, 1999

Table 22. ELS Configuration Command Summary (continued)

Command	Function
List	Lists information on ELS settings and messages.
Nodisplay	Disables message display on the console.
Noremote	Disables remote logging to a remote workstation.
Notrace	Controls disablement of packet trace events.
Notrap	Keeps messages from being sent out in SNMP traps.
Remote	Allows messages to be logged to a remote workstation.
Set	Sets the pin parameter and the timestamp feature 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.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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*

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

remote

display *event* **or** *group* **or** *subsystem*
Controls the output of the display of messages to the monitoring.
trap *event* **or** *group* **or** *subsystem*
Controls the generation of traps to the network management station.
remote *event* **or** *group* **or** *subsystem*
Controls the generation of traps to the remote station.

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*

Display

Use the **display** command to enable message displaying on the monitoring monitor for specific events, a range of events for a subsystem, groups, or subsystems.

Syntax:

display event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Displays messages of the specified event (*subsystem.event#*).

group *groupname*
Displays messages of a specified group (*groupname*).

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.

subsystem *subsystemname*
Displays messages associated with the specified subsystem. To find out which subsystems are on the device, type **list subsystems**.

Note: Although ELS supports all subsystems on the device, not all devices support all subsystems. See *Event Logging System Messages Guide* for a list of currently supported subsystems.

Filter

Use the **filter** command to access the filter configuration command environment. See “ELS Net Filter Configuration Commands” on page 127 for complete command details.

Syntax:

filter net

List

Use the **list** command to get updated information regarding ELS settings and listings of selected messages.

Syntax:

list all
filter-status
groups
pin
remote-log status
status
subsystem . . .
subsystems all
trace-status

all Lists information from all the **list** categories.

filter-status
Lists ELS net number filters.

groups
Lists the user-defined group names and contents.

pin Lists the current number of ELS event messages sent in SNMP traps (per second).

remote-log status
Lists the current values of remote logging options.

Example:

```
list r

Remote Logging is ON
Source IP Address = 192.67.38.2
Remote Log IP Address = 192.9.200.1
Default Syslog Facility = LOG_DAEMON
Default Syslog Priority Level = LOG_CRIT
Number of Messages in Remote Log = 256
Remote Logging Local ID = MYHOSTNAME
```

status Lists the subsystems, groups, and events that have been modified by the **display**, **nodisplay**, **trap**, **notrap**, **trace**, **notrace**, **remote**, and **noremote** commands.

Example:

```
list status

Subsystem:            TKR
```

```
Disp Levels:           STANDARD
Trap levels:           none
Trace levels:          none
Remote levels:         ERROR INFO TRACE
Syslog Facility/Level: LOG_USER LOG_INFO

Group        Disp     Trap    Trace  Remote
Mygroup     Unset    Unset    Unset    On
                                       Syslog Facilty/Level: LOG_DAEMON LOG_CRIT

Event        Disp     Trap    Trace  Remote
IP.007      Unset    Unset    Unset    On
                                       Syslog Facility/Level: LOG_CRON LOG_NOTICE
```

Note: Not only is remote logging enabled, but the display includes the Syslog Facility/Level values for each subsystem, group, and event. Ranges of events are listed as individual events.

subsystem

Lists names, events, and descriptions of all subsystems.

(Example output from a **list subsystem** command can be found beginning on page 134.)

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.

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
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
```

Nodisplay

Use the **nodisplay** command to select and turn off messages displaying on the console.

Syntax:

nodisplay event. . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Suppresses the displaying of a specified event (*subsystem.event#*).

group *groupname*
Suppresses the displaying of messages that were previously added to the specified group (*groupname*).

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.

subsystem *subsystemname*
Suppresses the displaying of messages associated with the specified subsystem.

Noremote

Use the **noremote** command to suppress the logging of events to a remote workstation based on event number, group, range of events, or subsystem.

Note: With the **noremote** command, there is usually no need to specify a *syslog_facility* and *syslog_level*, such as there is with the **remote** command. However, for **noremote subsystem** command, there exists the option of selectively suppressing specific message levels (for example, “error” only or “trace” only) rather than turning them all off. (If you do not specify any particular message level, “all” is assumed). Additionally, with the **noremote subsystem** command, you can set a *syslog_facility* and *syslog_level* for any remaining message levels that have not been turned off.

Syntax:

noremote event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Suppresses the remote logging of messages for the specified event.

group *group.name*
Suppresses the remote logging of messages that were previously added to the specified group (*group.name*).

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 remote logging of a range of messages for the specified subsystem.

Example:

```
noremote range gw 19 22
```

Suppresses the remote logging of events gw.019, gw.020, gw.021, and gw.022

subsystem *subsystem.name [syslog_facility syslog_level]*
Suppresses the remote logging of messages associated with the specified subsystem (*subsystem.name*).

Example 1:

```
noremote subsystem tkr
```

Suppresses the remote logging of all “tkr” messages.

Example 2:

```
ELS config> noremote subsystem tkr info
ELS config> SYSLOG FACILITY[LOG_USER]?
ELS config> SYSLOG LEVEL[LOG_INFO]?
```

In this example, “LOG_USER” and “LOG_INFO” were the values last picked for subsystem TKR. The command specified turns off the remote logging for subsystem TKR only for messages coded for “info”. Because *syslog_facility* and *syslog_level* was not specified, the software prompts for *syslog_facility* and *syslog_level*. If you enter another value at the prompts, that value will replace *syslog_facility* and *syslog_level* for the remaining remote-logged messages for the TKR subsystem.

Use the **list all** or **list status** commands to display what you have set with the **noremote** and **remote** commands.

For more information about *syslog_facility* and *syslog_level* see “Remote” on page 121.

Notrace

Disables packet trace for the specified event/range/subsystem/group.

Syntax:

notrace event . . .
group . . .
range . . .

subsystem . . .

event *subsystem.event#*
Suppresses the sending of packet trace data for the specified event#

group *groupname*
Suppresses the sending of packet trace data that was previously added to the specified group (groupname).

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:

```
trace range gw 19 22
```

Suppresses the sending of packet trace data for events gw.19, gw.20, gw.21, and gw.22.

subsystem *subsystemname*
Suppresses the sending of packet trace data for the specified subsystem (subsystemname).

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 . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Suppresses the sending of the specified message in an SNMP trap (*subsystem.event#*).

group *groupname*
Suppresses the sending of messages in SNMP traps that were previously added to the specified group (*groupname*).

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.

subsystem *subsystemname*
Suppresses the sending of messages in SNMP traps that are associated with the specified subsystem.

Remote

Use the **remote** command to select the events to be logged to a remote workstation by event number, range of events, group, or subsystem.

Syntax:

remote event . . .
range . . .
group . . .
subsystem . . .

event *subsystem.event# syslog_facility syslog_level*
Causes the specified event to be logged remotely.

Syslog facility and level values are used by the syslog daemon in the remote workstation to determine where to log the messages. This value overrides the default values that are set with the **set facility** and **set level** commands.

syslog_facility
log_auth
log_authpriv
log_cron
log_daemon
log_kern
log_lpr
log_mail
log_news
log_syslog
log_user
log_uucp
log_local0-7

syslog_level
log_emerg
log_alert
log_crit
log_err
log_warning
log_notice
log_info
log_debug

These values do NOT have any particular association with any daemons on the IBM 8371. They are merely identifiers which are used by the syslog daemon on the remote workstation.

range *subsystemname first_event_number last_event_number syslog_facility syslog_level*

Where *first_event_number* is the number of the first event in the specified event range, and *last_event_numbe*r is the number of the last event of the specified event range.

Causes the events in the specified range for the specified subsystem to be remotely logged based on the *syslog_facility* and *syslog_level* values. See "the remote event command" on page 121.

Example:

```
remote range gw 19 22 log_user log_info
```

Causes the event gw.19, gw.20, gw.21, and gw.22 to be logged remotely on the *syslog_facility* value of log_user and the *syslog_level* value of log_info.

group *group.name syslog_facility syslog_level*

Allows events belonging to the specified group to be logged remotely based on the *syslog_facility* and *syslog_level* values. See "the remote event command" on page 121.

subsystem *subsystem.name message_level syslog_facility syslog_level*

Where *subsystem.name* is the name of the subsystem and *message_level* is the level of messages selected in the subsystem.

Causes the events within the specified *subsystem.name* whose *message_level* agrees with the specified *message_level* to be logged remotely at the files based on the *syslog_facility* and *syslog_level* values. See "the remote event command" on page 121.

Message_level is a value such as "ALL," "ERROR," "INFO," or "TRACE". See "Logging Level" on page 99. The value specified in the **remote** command must agree with the value as coded on the particular event within the subsystem, or that event within the subsystem will not be remotely logged.

Example:

```
remote subsystem ETH all log_user log_info
```

In the above example, all messages in subsystem ETH ("all" includes any messages coded for "error," "info," or "trace") will be logged remotely based on log_user and log_info values at the remote host.

Use the **list all** or **list status** commands to display what you have set with the **noremote** and **remote** commands.

Set

Use the **set** command to set the maximum number of tags per second, the timestamp feature, or to set tracing options.

Syntax:

set pin . . .

remote-logging . . .

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.)

remote-logging

Use the **set remote-logging** command to configure remote logging options. When these options are configured from the monitoring environment, the changes take effect immediately, and return to their previously configured settings when the device is rebooted.

Syntax:

set remote-logging on
off
facility . . .
level . . .
no-msgs
remote_ip_addr . . .
source_ip_addr ...
local_id

on Turns remote logging on. Remote logging is now enabled to allow any messages selected by the **remote** command to be actively logged.

off Turns remote logging off. All messages selected by the 'remote' command will be prevented from being logged.

facility

Specifies a value that, in combination with the *level* value, is used by the syslog daemon in the remote workstation to determine where to log messages. This value is used for all remotely-logged ELS messages unless you specify a different value for a particular ELS event, range, group, or subsystem with the **remote** command.

These are all possible syslog facility values:

log_auth
log_authpriv
log_cron
log_daemon
log_kern
log_lpr
log_mail
log_news
log_syslog
log_user
log_uucp
log_local0-7

level Specifies a value that, in conjunction with the *facility* value, is used by the syslog daemon in the remote workstation to determine where to log messages. This value is used for all remotely-logged ELS

messages unless you specify a different value for a particular ELS event, range, group, or subsystem with the **remote** command.

These are all possible syslog level values:

log_emerg
log_alert
log_crit
log_err
log_warning
log_notice
log_info
log_debug

no-msgs
Specifies the number of messages in the buffer for the remote log before log wraps.

remote_ip_addr
This is an ip address of the form xxx.xxx.xxx.xxx where xxx can be any integer 0 to 255. It represents the ip address of the remote host where the log files reside.

source_ip_addr
This is an ip address of the form xxx.xxx.xxx.xxx where xxx can be any integer 0 to 255.

You should use an IP address that is configured in the 8371 for easier identification when the IP address or the hostname is shown in the remotely-logged ELS message. You should also verify that this IP address is quickly resolved to a hostname by the name server, or at least that the name server responds quickly with "address not found."

To determine that the IP address resolves properly enter the **host** command on your workstation as shown:

```
workstation>host 5.1.1.1
host: address 5.1.1.1 NOT FOUND
workstation>
```

If the response takes more than 1 second, select an IP address that resolves more quickly.

local_id
This is any character string of up to 32 characters, which is included in the logged message at the remote file and can help identify which machine logged the message.

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 device 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.

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. If you configure tracing options from the monitoring environment, the changes take effect immediately. They 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-pkt
- max-bytes-per-pkt
- memory-trace-buffer-size
- off
- on
- reset
- stop-event
- 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.

memory-trace-buffer-size *bytes*
Sets the size, in bytes, of the RAM trace buffer.

Valid Values: 0, ≥10,000

Default Value: 0

off Disables packet tracing.

on Enables packet tracing.

reset Clears the trace buffer and resets all associated counters.

stop-event *event id*
Stops tracing when an event (event id) occurs. Enter either an ELS event id (for example: TCP.013) or “None”. “None” is the default. Tracing stops only if the display of the particular ELS event is enabled.

When a stop-event occurs, an entry is written to the trace buffer. The **view** command for this trace entry will display “Tracing stopped due to ELS Event Id: TCP.013”.

After tracing stops due to a stop-event, you must re-enable tracing with the **set trace on** command. (A restart will also re-enable tracing if enabled from the `ELS Config>` prompt.)

wrap-mode [off or **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.

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 . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Causes the specified trace event (*subsystem.event#*) to be displayed on the system monitoring.

group *groupname*
Allows trace events that were previously added to the specified group to be displayed on the device monitoring.

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 monitoring.

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 monitoring.

subsystem *subsystemname*
Allows trace events associated with the specified subsystem to be displayed on the device monitoring.

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 . . .

group . . .

range . . .

subsystem . . .

event *subsystem.event#*
Causes the specified message (*subsystem.event#*) 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.

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.

subsystem *subsystemname*
Allows messages associated with the specified subsystem to be sent to a management station in an SNMP trap.

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 device.

ELS Net Filter Configuration Commands

ELS net filters give you the capability of looking only at ELS messages with certain net numbers and discarding other ELS messages.

When you create a filter, you specify the subsystem, event, or range of events to which the filter applies. You also specify the queue (for example, “DISPLAY”, “TRAP”, “TRACE”, or “REMOTE-LOGGING”). Finally, you specify the net number (or range of net numbers) that you want to filter.

When you enable the filter, messages that have been turned on by the ELS commands are subject to filtering. The filter allows only messages with the specified net numbers. The filter causes the device to discard messages that do not contain the specified net numbers.

By reducing the number of ELS messages sent, you can more easily locate messages for the interfaces in which you are interested.

This section describes the commands to configure the ELS net filters. To configure these filters, enter the **filter net** command at the ELS> prompt. Then, enter the configuration commands at the `ELS Filter net>` prompt.

Table 23. ELS Net Filter Configuration Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Create	Creates a filter and assigns it a number. A maximum of 64 filters is allowed.
Delete	Deletes a specified filter number or all filters.
Disable	Disables a specified filter number or all filters.
Enable	Enables a specified filter number or all filters.
List	Lists a specified filter number or all filters.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Create

Use the **create** command to create an ELS net filter.

Syntax:

create *queue* event *event_name net#_start net#_end*

range *event_range net#_start net#_end*

subsystem *subsystem_name net#_start net#_end*

queue The queue for which you are setting the filter. The valid queues are:

Display

Trace

Trap

Remote

event *event_name net#_start net#_end*

Specifies the event and net numbers that you are filtering.

If you specify *net#_start* and *net#_end* as the same number, you are filtering on a single net number.

The command **`create trap event GW.009 2 10`** filters traps for message GW.009 for net numbers 2 through 10.

range *event_range net#_start net#_end*

Specifies the range of ELS messages and net numbers that you are filtering.

If you specify *net#_start* and *net#_end* as the same number, you are filtering on a single net number.

The command **`create remote range ipx 19 22 3 6`** filters all ipx messages beginning with IPX.019 and ending with IPX.022 for net numbers 3 through 6 for remote logging.

subsystem *subsystem_name net#_start net#_end*

Specifies the subsystem and net numbers that you are filtering.

If you specify *net#_start* and *net#_end* as the same number, you are filtering on a single net number.

The command **`create display subsys ip 1 1`**, filters all ELS messages for the ip subsystem that contain net number 1 to the display. All other ip subsystem messages are discarded.

Delete

Use the **delete** command to delete a specific ELS filter or all ELS filters.

Syntax:

delete all
filter *filter#*

all Deletes all currently configured filters.

filter *filter#*
Deletes the filter specified by *filter#*. Use the **list** command to obtain the number for the filter you want to delete.

Disable

Use the **disable** command to disable a specific ELS filter or all ELS filters.

Syntax:

disable all
filter *filter#*

all Disables all currently configured filters.

filter *filter#*
Disables the filter specified by *filter#*. Use the **list** command to obtain the number for the filter you want to disable.

Enable

Use the **enable** command to enable a specific ELS filter or all ELS filters.

Syntax:

enable all
filter *filter#*

all Enables all currently configured filters.

filter *filter#*
Enables the filter specified by *filter#*. Use the **list** command to obtain the number for the filter you want to enable.

List

Use the **list** command to list a specific ELS filter or all ELS filters.

Syntax:

list all
filter *filter#*

all Lists all currently configured filters.

filter Lists the filter specified by *filter#.*

Entering and Exiting the ELS Operating Environment

The ELS monitoring 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 13. Configuring and Monitoring the Event Logging System (ELS)” on page 113.

To enter the ELS monitoring environment from OPCON:

1. Enter the **console** command.

   ```
   * console
   ```

   The monitoring displays the GWCON prompt (+). If the prompt does not appear when you first enter GWCON, press **enter**.
2. At the GWCON prompt, enter the following command to access ELS:

   ```
   + event
   ```

   The monitoring displays the ELS monitoring prompt (ELS>). Now, you can enter ELS monitoring commands.

To leave the ELS monitoring environment, enter the **exit** command.

ELS Monitoring Commands

This section summarizes and then explains all the ELS monitoring commands. After accessing the ELS Monitoring environment, you can enter ELS monitoring commands at the ELS> prompt.

Table 24. ELS Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Clear	Resets to zero the counts of messages associated with specified events, groups, or subsystems.
Display	Enables message display on the console.
Exit	Exits the ELS console process and returns the user to GWCON.
Filter	Filter ELS messages based upon the net number.
List	Lists information on ELS settings and messages.
Nodisplay	Disables message display on the console.
Noremote	Disables remote logging to file at remote workstation.
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.
Remote	Allows messages to be logged at a file on a remote workstation.
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.

Table 24. ELS Monitoring Command Summary (continued)

Command	Function
Trap	Allows messages to be sent to a network management workstation in SNMP traps.
View	Allows viewing of traced packets.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Clear

Use the **clear** command to reset to zero the counts of the display, trace, trap, or remote commands as they relate to specific events, groups or subsystems.

Syntax:

clear event . . .
group . . .
subsystem . . .

event *subsystem. event#*
Resets the count of events to zero for displaying, trapping, tracing or remote logging of the specified event (*subsystem.event#*).

group *group.name*
Resets the count of events to zero for displaying, trapping, tracing or remote logging of the specified group (*group.name*).

subsystem *subsystem.name*
Resets the count of events to zero for displaying, trapping, tracing or remote logging of the specified subsystem (*subsystem.name*).

Display

Use the display command to enable the message display on the monitoring monitor for specific events.

Syntax:

display event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Displays messages for the specified event (subsystem.event#).

group *groupname*
Displays messages of a specified group (*groupname*).

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.

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.

Files Trace TFTP

Use the **files trace tftp** command to retrieve trace files from the subdirectory associated with:
- The currently active bank (bank A or bank B on the hard disk)
- The trace file stored in the Network Subdirectory (if there is no active bank)

Syntax:

files trace tftp active-bank ...
bank-a ...
bank-b ...
net-subdir ...

You are prompted for the *remote server IP address* and the *remote path/file name*.

active-bank
Retrieves the traces file from the currently active bank

bank-a
Retrieves the trace file from bank A

bank-b
Retrieves the trace file from bank B

net-subdir
Retrieves the trace file stored in the Network Subdirectory (if there is no active bank)

Files

Use the **files** command to transfer trace files to another host on the network using TFTP.

Syntax:

files trace tftp *host_IP_addr filename*

host_IP_addr
Is the IP address of the host to which you are transferring the files.

filename
Is the target file name. For TFTP, the file name must be fully path specified, and the file name must already exist on the target host.

Filter

Use the **filter** command to access the filter configuration command environment. See “ELS Net Filter Monitoring Commands” on page 151 for complete command details.

Syntax:

filter net

List

Use the **list** command to get updated information regarding ELS settings and to get listings of selected messages.

Syntax:

list all
active . . .
event . . .
filter-status
groups . . .
pin
remote-log status
subsystem . . .
trace-status

all Lists all subsystems, defined groups, enabled subsystems, enabled events, and pins.

active *subsystem.name*
Displays the events that are active for a specific subsystem or have non-zero message counts.

Example:

```
list active ip
Event      Active  Count  Message

IP.007              2874  %I -> %I
IP.022                13  add nt %I int %I nt %n int %s/%d
IP.036              2874  rcv pkt prt %d frm %I
IP.058                23  del nt %I rt via %I nt %n int %s/%d
IP.068     D          37  routing cache cleared
D=Display on   T=Trap on   P=Packet Trace on   F=Filter on  R=Remote Logging on
A=Advanced on
```

If Remote logging is turned on, those events displayed as active for a subsystem will have an “R” next to their name.

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
```

If Remote-logging had been activated for this event, and the *syslog_facility* and *syslog_level* values were log_daemon and log_crit, the last lines would look like:

```
Active:  R count:84182
Syslog Facility: log_daemon   Syslog Level: log_crit
```

filter-status
Lists ELS net number filters.

groups *group.name*
Displays the user-defined group names.

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
```

remote-log status
Lists the current values of the remote logging options set in the **set remote-logging** command.

Example:

```
list r

Remote Logging is On
Source Ip Address = 192.9.200.8
Remote  Log IP Address = 192.9.200.1
Default Syslog Facility = LOG_USER
Default Syslog Priority Level = LOG_INFO
Number of Messages in Remote Log = 256
Remote Logging Local ID =  SPHINX
```

subsystem *subsystem.name*
Lists event names, the total number of events that have occurred, and their descriptions.

Note: Although ELS supports all subsystems on the device, not all devices support all subsystems. See *ELS Messages* for a list of currently supported subsystems.

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
```

subsystem all
Lists all events, logging levels, and messages for every event that has occurred on the device.

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
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

------------------------ Run-time Status ---------------------------
Packets in RAM Trace Buffer:1   Free Trace Buffer Memory:99958
Trace Errors:0  First Packet:1  Last Packet:1
Trace Records Stored on HD:8  Trace Buffer File Size:16560
HD-Shadowing Time Exceeded? NO Elapsed Time:  0 hr,  0 min, 10 sec
Has Stop Trace Event Occurred? NO
```

- “Trace Status” in the LIST TRACE-STATUS display will indicate OFF when STOP-ON-EVENT action occurs.
- “Trace Buffer File Size” will display <wrapped> when a wraparound has occurred in the trace file.

ELS Config>**LIST TRACE** command under **talk 6** displays information similar to the following:

```
------------------------ Configuration -----------------------------
Trace Status:ON  Wrap Mode:ON  Decode Packets: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
```

Nodisplay

Use the **nodisplay** command to select and turn off messages displaying on the console.

Syntax:

nodisplay event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*

Suppresses the displaying of messages for the specified event.

group *group.name*

Suppresses the displaying of messages that were previously added to the specified group (*group.name*).

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.

subsystem *subsystem.name*
Suppresses the displaying of messages associated with the specified subsystem (*logging level*).

Noremote

Use the **noremote** command to select and turn off messages logging to a remote workstation.

Syntax:

noremote event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Suppresses the remote logging of messages for the specified event.

group *group.name*
Suppresses the remote logging of messages that were previously added to the specified group (*group.name*).

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 remote logging of a range of messages for the specified subsystem.

Example:

```
noremote range gw 19 22
```

Suppresses the remote logging of events gw.19, gw.20, gw.21, and g.22

subsystem *subsystem.name*
Suppresses the remote logging of messages associated with the specified subsystem (*logging level*).

Example:

```
noremote subsystem tkr
```

Note: With noremote, there is no need to specify a Syslog Facility and Level, such as there is with Remote.

Use the **list event** and **list active** commands to verify what you set with the **remote** and **noremote** commands.

Notrace

Use the **notrace** command to stop display of selected trace events at the monitoring.

Syntax:

notrace event . . .

group . . .

range . . .

subsystem . . .

event *subsystem.event#*
Suppresses the display of the specified tracing event.

group *groupname*
Suppresses the display of tracing events related to the specified group (*groupname*).

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.

subsystem *subsystemname [logging-level]*
Suppresses the display of tracing events that are associated with the specified subsystem and logging level. If you do not specify a *logging-level* you suppress tracing for all logging levels for the subsystem.

Example:

```
notrace subsystem frl error

notrace subsystem frl
```

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. . .

group . . .

range . . .

subsystem . . .

event *subsystem.event#*
Suppresses the sending of the specified message in an SNMP trap (*subsystem.event#*).

group *groupname*
Suppresses the sending of messages in SNMP traps that were previously added to the specified group (*groupname*).

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.

subsystem *subsystemname [logging-level]*

Suppresses the sending of messages in SNMP traps that are associated with the specified subsystem and logging level. If you do not specify a *logging-level* you suppress trapping for all logging levels for the subsystem.

Example:

```
notrap subsystem eth error
```

Packet Trace

Use the **packet-trace** command to display/enable/disable packet tracing information for various subsystems.

Syntax:

packet-trace

Use the **Exit** command when you are finished using Packet Trace.

For complete command descriptions, see “Packet-trace Monitoring Commands” on page 148.

Remote

Use the **remote** command to select the events to be logged to a remote file by event number, range of events, group, or subsystem.

Syntax:

remote event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event# syslog_facility syslog_level*

Causes the specified event to be logged remotely.

Syslog facility and level values are used by the syslog daemon in the remote workstation to determine where to log the messages. This value overrides the default values that are set with the **set facility** and **set level** commands.

syslog_facility

log_auth

log_authpriv
log_cron
log_daemon
log_kern
log_lpr
log_mail
log_news
log_syslog
log_user
log_uucp
log_local0-7

syslog_level

log_emerg
log_alert
log_crit
log_err
log_warning
log_notice
log_info
log_debug

These values do NOT have any particular association with any daemons on the IBM 8371. They are merely identifiers which are used by the syslog daemon on the remote workstation.

Example:

```
remote event gw.019 log_user log_info
```

group *group.name syslog_facility syslog_level*
Allows events belonging to the specified group to be logged remotely based on the *syslog_facility* and *syslog_level* values. See “the remote event command” on page 138.

range *subsystemname first_event_number last_event_number syslog_facility syslog_level*

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 events in the specified range for the specified subsystem to be remotely logged based on the *syslog_facility* and *syslog_level*. See “the remote event command” on page 138.

Example:

```
remote range gw 19 22 log_user log_info
```

Causes the event gw.19, gw.20, gw.21, and gw.22 to be logged remotely to the files specified by the *syslog_facility* value of log_user and the *syslog_level* value of log_info.

subsystem *subsystem.name message_level syslog_facility syslog_level*
Where *subsystem.name* is the name of the subsystem and *message_level* is the level of messages selected in the subsystem.

Causes the events within the specified *subsystem.name* whose *message_level* agrees with the specified *message_level* to be logged remotely based on the *syslog_facility* and *syslog_level*. See “the remote event command” on page 138.

Message_level is a value such as ALL, ERROR, INFO, or TRACE. See “Logging Level” on page 99. The value specified in the **remote** command must agree with the value as coded on the particular event within the subsystem, or that event within the subsystem will not be remotely logged.

Example:

```
remote subsystem eth all log_user log_info
```

In the above example, all messages in subsystem TKR (“all” includes any messages coded for “error,” “info,” or “trace”) will be logged remotely to files specified by log_user and log_info at the remote host.

Use the **list event** and **list active** commands to verify what you set with the **remote** and **noremote** commands.

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

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

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

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 monitoring

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

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.

Syntax:

set pin . . .
remote-logging . . .
timestamp . . .
trace . . .

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.)

remote-logging
Use the **set remote-logging** command to configure remote logging options. When these options are configured from the monitoring environment, the changes take effect immediately, and return to their previously configured settings when the device is rebooted.

Syntax:

set remote-logging on
off
facility . . .
level . . .
local_id
remote_ip_addr . . .
source_ip_addr ...

on Turns remote logging on. Remote logging is now enabled to allow any messages selected by the **remote** command to be actively logged.

off Turns remote logging off. All messages selected by the **remote** command will be prevented from being logged.

facility
Specifies a value that, in combination with the *level* value, is used by the syslog daemon in the remote workstation to determine where to log messages. This value is used for all remotely-logged ELS messages unless you specify a different value for a particular ELS event, range, group, or subsystem with the **remote** command.

These are all possible syslog facility values:

log_auth
log_authpriv
log_cron
log_daemon
log_kern
log_lpr
log_mail
log_news
log_syslog
log_user
log_uucp
log_local0-7

level Specifies a value that, in conjunction with the *facility* value, is used by the syslog daemon in the remote workstation to determine where to log messages. This value is used for all remotely-logged ELS messages unless you specify a different value for a particular ELS event, range, group, or subsystem with the **remote** command.

These are all possible syslog level values:

log_emerg
log_alert
log_crit
log_err
log_warning
log_notice
log_info
log_debug

local_id
Specifies a 1-32 character identifier that appears in the remote logging message that you can use to identify which machine logged a particular message.

remote_ip_addr
This is an IP address of the remote host where the log files reside.

source_ip_addr
Specifies the IP address of the machine that originated the message that is being remotely-logged.

You should use an IP address that is configured in the 8371 for easier identification when the IP address or the hostname is shown in the remotely-logged ELS message. You should also verify that this IP address is quickly resolved to a hostname by the name server, or at least that the name server responds quickly with "address not found."

To determine that the IP address resolves properly enter the **host** command on your workstation as shown:

```
workstation>host 5.1.1.1
host: address 5.1.1.1 NOT FOUND
workstation>
```

If the response takes more than 1 second, select an IP address that resolves more quickly.

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 device 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 device'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 device. 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]

trace Use the **set trace** command to configure tracing options. When tracing options are configured from the monitoring environment, the changes take effect immediately, and return to their previously configured settings when the device is rebooted.

Syntax:

set trace	decode . . .
	default-bytes-per-pkt . . .
	max-bytes-per-pkt . . .
	memory-trace-buffer-size . . .
	off
	on
	reset
	stop-event . . .
	wrap-mode . . .

decode . . .

Sets packet decode options. Packet decoding is not supported by all components.

off Sets decoding off.

on Sets decoding on.

Note: The default setting is to print complete decode output for all frame types. Use the **list trace-status** command to see the current decode settings. See page 134.

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.

memory-trace-buffer-size *bytes*
Sets the size, in bytes, of the RAM trace buffer.

Valid Values: 0, ≥10,000

Default Value: 0

off Disables packet tracing.

on Enables packet tracing.

reset Clears the trace buffer and resets all associated counters.

stop-event *event id*
Stops tracing when an event (event id) occurs. Enter either an ELS event id (for example: TCP.013) or “None”. “None” is the default. Tracing stops only if the display of the particular ELS event is enabled.

When a stop-event occurs, an entry is written to the trace buffer. The **view** command for this trace entry will display “Tracing stopped due to ELS Event Id: TCP.013”.

After tracing stops due to a stop-event, you must re-enable tracing with the **set trace on** command. (A restart will also re-enable tracing if enabled from the `ELS Config>` prompt.)

Example:

```
set trace stop-event TCP.013
```

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.

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 monitoring. This command provides function that is similar to the **packet trace** command described in “Packet-trace Monitoring Commands” on page 148.

Syntax:

trace event . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Causes the specified trace event (*subsystem.event#*) to be displayed on the system monitoring.

group *groupname*
Allows trace events that were previously added to the specified group to be displayed on the device monitoring.

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_numbe*r 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 monitoring.

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 monitoring.

subsystem *subsystemname*
Allows trace events associated with the specified subsystem to be displayed on the device monitoring.

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 . . .
group . . .
range . . .
subsystem . . .

event *subsystem.event#*
Causes the specified message (*subsystem.event#*) 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.

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.

subsystem *subsystemname*
Allows messages associated with the specified subsystem to be sent to a management station in an SNMP trap.

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 device.

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
Displays the next traced packet that contains the specified information. You can specify the search information by:

- Hexadecimal string
- IP address
- ASCII text

Packet-trace Monitoring Commands

This section describes the Packet-trace Monitoring commands. After accessing the Packet-trace Monitoring environment, you can enter Packet-trace Monitoring commands at the `ELS Packet Trace>` prompt.

Table 25. Packet Trace Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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 subsystems that support packet tracing, or displays a summary.
Trace-status	Displays information on the status of packet tracing, including configuration and run-time.
View	Provides View Captured Packet Trace Buffers Console
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Off

Use the **off** command to disable packet tracing.

Syntax:

off

On

Use the **on** command to enable packet tracing.

Syntax:

on

Reset

Use the **reset** command to clear the trace buffer and reset all associated counters.

Syntax:

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

For an explanation of the `set` command, see page 143.

Subsystems

Use the **subsystems** command to activate tracing for the subsystems that support packet tracing, or to display a summary.

Syntax:

subsystems	atm
	lec
	summary

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.

MAC Address packet filtering can be enabled under the LEC net
using the 'trace mac-address' command.
```

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
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

----------------------- 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

Has Stop Trace Event Occurred? NO
```

View

Use the **view** command to enter the View Captured Packet Trace Buffers Monitoring.

For an explanation of the **view** commands, see “View” on page 147.

Syntax:

view

- current
- first
- jump
- last
- next
- prev
- search

exit

ELS Net Filter Monitoring Commands

This section describes explains the commands to manipulate ELS net filters. To enter the filter environment, enter the **filter net** command at the ELS> prompt. Enter the monitoring commands at the `ELS Filter net>` prompt.

Table 26. ELS Net Filter Monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Create	Creates a filter and assigns it a number. A maximum of 64 filters is allowed.
Delete	Deletes a specified filter number or all filters.
Disable	Disables a specified filter number or all filters.
Enable	Enables a specified filter number or all filters.
List	Lists a specified filter number or all filters.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Create

Use the **create** command to create an ELS net filter.

Syntax:

create *queue* event *event_name net#_start net#_end*

range *event_range net#_start net#_end*

subsystem *subsystem_name net#_start net#_end*

queue The queue for which you are setting the filter. The valid queues are:

Display

Trace

Trap

Remote

event *event_name net#_start net#_end*

Specifies the event and net numbers that you are filtering.

If you specify *net#_start* and *net#_end* as the same number, you are filtering on a single net number.

The command **`create trap event GW.009 2 10`** filters traps for message GW.009 for net numbers 2 through 10.

range *event_range net#_start net#_end*

Specifies the range of ELS messages and net numbers that you are filtering.

If you specify *net#_start* and *net#_end* as the same number, you are filtering on a single net number.

The command **`create remote range ipx 19 22 3 6`** filters all ipx messages beginning with IPX.019 and ending with IPX.022 for net numbers 3 through 6 for remote logging.

subsystem *subsystem_name net#_start net#_end*
Specifies the subsystem and net numbers that you are filtering.

If you specify *net#_start* and *net#_end* as the same number, you are filtering on a single net number.

The command **create display subsys ip 1 1**, filters all ELS messages for the ip subsystem that contain net number 1 to the display. All other ip subsystem messages are discarded.

Delete

Use the **delete** command to delete a specific ELS filter or all ELS filters.

Syntax:

delete all
filter *filter#*

all Deletes all currently configured filters.

filter *filter#*
Deletes the filter specified by *filter#*. Use the **list** command to obtain the number for the filter you want to delete.

Disable

Use the **disable** command to disable a specific ELS filter or all ELS filters.

Syntax:

disable all
filter *filter#*

all Disables all currently configured filters.

filter *filter#*
Disables the filter specified by *filter#*. Use the **list** command to obtain the number for the filter you want to disable.

Enable

Use the **enable** command to enable a specific ELS filter or all ELS filters.

Syntax:

enable all
filter *filter#*

all Enable all currently configured filters.

filter *filter#*
Enable the filter specified by *filter#*. Use the **list** command to obtain the number for the filter you want to enable.

List

Use the **list** command to list a specific ELS filter or all ELS filters.

Syntax:

list all
filter *filter#*

all Lists all currently configured filters.

filter *filter#*
Lists the filter specified by *filter#*.

Chapter 14. Configuring and Monitoring Performance

This chapter describes how to use the Performance configuration and monitor operating commands and includes the following sections:

Performance Overview

Configuring performance allows you to monitor your CPU load. In the idle (non-work load) state, performance reflects operations that the device continuously performs as a part of managing external interfaces. The CPU load registered in the idle state is dependent upon:

- Number of protocols running.
- Number of interfaces/cards installed.
- Type of interfaces installed.

The performance function can be used as a tool for trend analysis, bottleneck evaluation, and capacity planning. By collecting the CPU utilization information on the device, a network manager can monitor:

- CPU load versus time of day.
- CPU load versus location of the device in the network.
- CPU load versus traffic throughput.
- CPU load versus user load

Performance Reporting Accuracy

If you request a performance analysis when the 8371 first comes online, you will see values that reflect an initialization state that has little or no network traffic, so it is of little use in helping to balance your network load.

It is best to use performance reports that are generated under normal loads after approximately 2 minutes of operation.

Accessing the Performance Configuration Environment

Use the following procedure to access the Performance monitor configuration process.

1. At the OPCON prompt, enter **talk 6**. (For more detail on this command, see "What is CONFIG?" on page 53.) For example:

```
* talk 6
Config>
```

© Copyright IBM Corp. 1996, 1999

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

2. At the CONFIG prompt, enter the **perf** command to get to the `PERF Config>` prompt.

Performance Configuration Commands

To configure Performance, enter the commands at the `PERF Config>` prompt.

Table 27. PERF Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Disable	Disables the collection of CPU utilization statistics or `Talk 2` ELS monitor output.
Enable	Enables the collection of CPU utilization statistics or `Talk 2` ELS monitor output.
List	Lists the configuration.
Set	Sets the reporting period.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Disable

Use the **disable** command to disable collection of CPU utilization statistics and disable the `talk 2` ELS monitor output.

Syntax:

disable	cpu statistics t2 output

Enable

Use the **enable** command to enable collection of CPU utilization statistics and enable the `talk 2` ELS monitor output.

Syntax:

enable	cpu statistics t2 output

List

Use the **list** command to display the performance monitor configuration.

Syntax:

list

Set

Use the **set** command to set the reporting period.

Syntax:

set	*time*

time Specifies the short window time.

Valid Values: 2 - 30 seconds

Default Value: 5

Accessing the Performance Monitoring Environment

Use the following procedure to access the Performance monitoring commands. This process gives you access to the Performance *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. (For more detail on this command, see "Chapter 10. The Operating/Monitoring Process (GWCON - Talk 5) and Commands" on page 83.) For example:

```
* talk 5
+
```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **enter** again.

2. At the + prompt, enter the **perf** command to get you to the `PERF Console>` prompt.

   Example:

```
+ perf
PERF Console>
```

Performance Monitoring Commands

This section describes the Performance monitoring commands.

Table 28. PERF Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Clear	Clear the CPU utilization high water statistics and resets the reporting period to a new cycle.
Disable	Disables the collection of CPU utilization statistics or `Talk 2` ELS monitor output.
Enable	Enables the collection of CPU utilization statistics or `Talk 2` ELS monitor output.
List	Lists the configuration.
Report	Displays a report of performance statistics.
Set	Sets the reporting period.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

Disable

Use the **disable** command to disable collection of CPU utilization statistics and disable the `talk 2` ELS monitor output.

Syntax:

disable cpu statistics
t2 output

Enable

Use the **enable** command to enable collection of CPU utilization statistics and enable the `talk 2` ELS monitor output.

Syntax:

enable cpu statistics
t2 output

List

Use the **list** command to display the performance monitor configuration.

Syntax:

list

Report

Use the **report** command to display performance monitor statistics.

Syntax:

report

Example:

```
PERF Console>report
------------------------------------------------------------
KEY:  SW = Short Window = 9 seconds
KEY:  LW = Long Window =  9.0 minutes  (60 x SW)

  CPU UTIL :   Most recent SW                    = 38%
               Most recent LW                    = 33%
               Highest for all SW's              = 92%
               Highest for all LW's              = 52%
               % of time cpu util (SW) was > 60%  = 16%
               % of time cpu util (SW) was > 70%  = 15%
               % of time cpu util (SW) was > 80%  = 1%
               % of time cpu util (SW) was > 90%  = 0%
               % of time cpu util (SW) was > 95%  = 0%
------------------------------------------------------------
```

Set

Use the **set** command to set the reporting period.

Syntax:

set *time*

time Specifies the short window time.

Valid Values: 2 - 30 seconds

Default Value: 5

Part 2. Interfaces

© Copyright IBM Corp. 1996, 1999

Chapter 15. Using the 10/100-Mbps Ethernet Network Interface

This chapter describes how to use the 10/100-Mbps Ethernet interface. It includes the following section:

- "Displaying 10/100-Mbps Ethernet Statistics"
- "Auto-negotiation on the 10/100-Mbps Ethernet Interface" on page 164

Displaying 10/100-Mbps Ethernet Statistics

You can use the **interface** command from the GWCON environment to display the following statistics.

```
+interface
                                                                Self-Test  Self-Test Maintenance
Net   Net'   Interface   Slot-Port                                 Passed     Failed      Failed
0     0      Eth/0       Slot: 0   Port: 1                            0          0           0
1     1      Eth/1       Slot: 0   Port: 2                            0          0           0
2     2      Eth/2       Slot: 0   Port: 3                            0          0           0
3     3      Eth/3       Slot: 0   Port: 4                            0          0           0
4     4      Eth/4       Slot: 0   Port: 5                            0          0           0
5     5      Eth/5       Slot: 0   Port: 6                            0          0           0
6     6      Eth/6       Slot: 0   Port: 7                            0          0           0
7     7      Eth/7       Slot: 0   Port: 8                            0          0           0
8     8      Eth/8       Slot: 0   Port: 9                            0          0           0
9     9      Eth/9       Slot: 0   Port: 10                           0          0           0
10    10     Eth/10      Slot: 0   Port: 11                           0          0           0
11    11     Eth/11      Slot: 0   Port: 12                           0          0           0
12    12     Eth/12      Slot: 0   Port: 13                           0          0           0
13    13     Eth/13      Slot: 0   Port: 14                           0          0           0
14    14     Eth/14      Slot: 0   Port: 15                           0          0           0
15    15     Eth/15      Slot: 0   Port: 16                           0          0           0
16    16     Eth/16      Slot: 1   Port: 1                            0          0           0
17    17     Eth/17      Slot: 1   Port: 2                            0          0           0
18    18     Eth/18      Slot: 1   Port: 3                            0          0           0
19    19     Eth/19      Slot: 1   Port: 4                            0          0           0
20    20     Eth/20      Slot: 1   Port: 5                            0          0           0
21    21     Eth/21      Slot: 1   Port: 6                            0          0           0
22    22     Eth/22      Slot: 1   Port: 7                            0          0           0
23    23     Eth/23      Slot: 1   Port: 8                            0          0           0
24    24     Eth/24      Slot: 2   Port: 1                            0          0           0
25    25     Eth/25      Slot: 2   Port: 2                            0          0           0
26    26     Eth/26      Slot: 2   Port: 3                            0          0           0
27    27     Eth/27      Slot: 2   Port: 4                            0          0           0
28    28     Eth/28      Slot: 2   Port: 5                            0          0           0
29    29     Eth/29      Slot: 2   Port: 6                            0          0           0
30    30     Eth/30      Slot: 2   Port: 7                            0          0           0
31    31     Eth/31      Slot: 2   Port: 8                            0          0           0
32    32     Eth/32      Slot: 3   Port: 1                            0          0           0
33    33     Eth/33      Slot: 3   Port: 2                            0          0           0
34    34     Eth/34      Slot: 3   Port: 3                            0          0           0
35    35     Eth/35      Slot: 3   Port: 4                            0          0           0
36    36     ATM/0       Slot: 1   Port: 4                            0          1           0
37    37     ATM/1       Slot: 1   Port: 2                            0          1           0
38    38     ATM/2       Slot: 2   Port: 1                            0          1           0
39    39     ATM/3       Slot: 2   Port: 2                            0          1           0
40    40     Eth/36                                                   0          0           0
41    41     Eth/37                                                   0          0           0
42    42     Eth/38                                                   0          0           0
43    43     Eth/39                                                   0          0           0
44    44     Eth/40                                                   0          0           0
45    55     Eth/42                                                   0          0           0
46    56     Eth/43                                                   0          0           0
47    57     Eth/44                                                   0          0           0
48    58     Eth/45                                                   0          0           0
49    59     Eth/46                                                   0          0           0
50    50     Eth/47                                                   0          0           0
51    51     Eth/48                                                   0          0           0
52    52     Eth/49                                                   0          0           0
53    53     Eth/50                                                   0          0           0
54    54     Eth/51                                                   0          0           0
55    65     Eth/52                                                   0          0           0
```

© Copyright IBM Corp. 1996, 1999

```
1                    56    66     Eth/53                          0          0          0
1                    57    67     Eth/54                          0          0          0
1                    58    68     Eth/55                          0          0          0
1                    59    69     Eth/56                          0          0          0
1                    60    60     Eth/57                          0          0          0
1                    61    61     Eth/58                          0          0          0
1                    62    62     Eth/59                          0          0          0
1                    63    63     Eth/60                          0          0          0
1                    +
```

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.

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.

Actual address

Configured duplex
The value configured for duplex. Values can be Half Duplex, Full Duplex, or Auto-Negotiation.

Actual duplex
The value at which the adapter is presently operating. It might be different from the value configured, depending on the switch capability. If the adapter is not Up, the value displayed will be *Unknown*. Otherwise the value can be Half Duplex or Full Duplex.

Whenever the link partner (switch or hub) does not participate during the negotiation phase, *** will follow the actual duplex mode value. When *** is indicated the operational duplex value should be verified on the switch or hub for consistency.

Most hubs (unlike switches) can only support half-duplex mode, and are not capable of negotiation. As such the *** indication will usually be displayed when the interface is connected to a hub.

A message will also be logged via the ELS system whenever a possibility of a mis-match in duplex mode exists.

Note: If the link partner (switch or hub) to which the interface is connected does not respond during the negotiation phase, the two may result in operating in different duplex modes. That is, the interface may be operating in half-duplex, while the switch port is operating in full duplex mode. A mismatch in the duplex mode can result in severe

performance degradation. See “10/100-Mbps Ethernet Configuration Commands” on page 165 for important information regards speed and duplex configurations.

Configured speed
The value configured for speed. Values can be 10 Mbps, 100Mbps, or Auto-Negotiation.

Actual speed
The speed at which the adapter is presently operating.If the adapter is not Up, the value displayed will be *Unknown*. Otherwise the value can be 10 Mbps or 100 Mbps.

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, alignment error
The Failed, Framing Error counter increments when the interface receives a packet where the length in bits is not a multiple of eight.

failed, receive overflow
Overflow error indicates that the receiver has lost all or part of the incoming frame, due to an inability to move data from the receive FIFO into memory buffer before the internal FIFO overflowed.

receive underrun
Indicates the number of times the adapter did not have a second buffer to store a long frame (requiring more than one buffer).

Output statistics:

one retry
Indicates that exactly one retry was needed to transmit a frame. 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.

failed, transmit underflow
Transmit underrun indicates that transmitter has truncated a message because it could not read data from the memory fast enough. It also indicates that the FIFO on the adapter has emptied out before the end of the frame was reached. IFO into memory buffer before the internal FIFO overflowed.

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.

memory error
Memory errors occur when the adapter is not given access to the system interface bus within the programmable length of time. This error will normally occur during transmit operations, indicating transmit underrun.

Auto-negotiation on the 10/100-Mbps Ethernet Interface

Specifying values other than *auto* for speed or duplex on the 10/100 Ethernet interface or its link partner (switch port) can result in duplex mode mismatch or link activation failures.

Link activation failures due to configuration mismatches will occur on the IBM 8371 whenever the speed configured at both ends are not identical.

When either speed or duplex value is *auto-negotiate*, both speed and duplex will be negotiated with the link partner and its configured speed or duplex will be used.

Chapter 16. Configuring and Monitoring the 10/100-Mbps Ethernet Network Interface

This chapter describes the 10/100-Mbps Ethernet interface configuration and operational commands. It includes the following sections:

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 **configuration**. (For more detail on this command, refer to "What is the OPCON Process?" on page 41.) For example:

   ```
   * configuration
   Config>
   ```

   The CONFIG prompt (`Config>`) displays on the console. If the prompt does not appear when you first enter configuration, press **enter** again.
2. At the CONFIG prompt, enter the **list devices** command to display the network interface numbers for which the device is currently configured.
3. Record the interface numbers.
4. Enter the **network** command and the number of the Ethernet interface you want to configure. For example:

   ```
   Config> network 0
   Ethernet 100 interface configuration
   ETH100 Config>
   ```

   The 10/100-Mbps Ethernet configuration prompt (`ETH100 Config>`), is displayed.

10/100-Mbps Ethernet Configuration Commands

This section describes the 10/100-Mbps Ethernet configuration commands. Enter the commands at the `ETH config>` prompt.

Table 29. 10/100-Mbps Ethernet Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Duplex	Sets the duplex mode.
IP-Encapsulation	Sets the IP encapsulation as Ethernet (type X'0800') or IEEE (802.3 with SNAP).
List	Displays the current connector-type, and IP encapsulation.
Physical-Address	Sets the physical MAC address.
Speed	Sets the link speed.

© Copyright IBM Corp. 1996, 1999

Table 29. 10/100-Mbps Ethernet Configuration Command Summary (continued)

Command	Function
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Duplex

Use the **duplex** command to set the duplex mode.

Note: The default value of *auto* is recommended. The value **half-duplex** or **full-duplex** should be specified only if auto-negotiation does not result in successful activation of the interface or desired duplex mode. Note when you see the command syntax that the command for half-duplex or full-duplex is written with an underline between the words, for example, *half_duplex*.

If a value other than *auto* is specified, ensure that the same value is configured on the switch port. After configuring the switch port to match the duplex specified on the 10/100-Mbps Ethernet interface, disable and test the interface.

Verify that the actual duplex mode shown on the interface status panel matches the operational value on the switch port.

The interface may enter the Up state with mismatched duplex mode. Operating with mismatched duplex modes on the interface and switch port can cause severe performance degradation.

Syntax:

duplex half_duplex
full_duplex
auto

Half_duplex
The interface will not transmit while receiving or receive while transmitting.

Full_duplex
The interface will transmit and receive simultaneously.

Auto The interface will automatically select half duplex or full duplex depending on the link partner’s capability.

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** for the type.

Syntax:

IP-encapsulation *type*

Example: `IP-encapsulation e`

List

Use the **list** command to display the current configuration for the 10/100-Mbps Ethernet interface.

Syntax:

list all

Example:

```
list all

The duplex is   HALF DUPLEX
The speed is   100Mb
IP Encapsulation:   Ether

MAC Address:        023456789A56
```

Physical-Address

Use the **physical-address** command to set the physical (MAC) address.

Syntax:

physical-address ***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 device 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:

```
physical-address
MAC address in 00:00:00:00:00:00 form []? 12:15:00:FA:00:FE
```

Speed

Use the **speed** command to set the speed used by this interface.

Note: The default value of *auto* is recommended. The values of **ten** and **hundred** should be specified only if auto-negotiation does not result in sucessful activation of the interface or desired speed.

If a value other than *auto* is specified, ensure that the same value is configured on the switch port. After configuring the switch port to match the speed specified on the 10/100-Mbps Ethernet interface, disable and test the interface.

If the interface and switch (or hub) port are not configured for identical speed, the interface will not attain the Up state.

See “Auto-negotiation on the 10/100-Mbps Ethernet Interface” on page 164 for information about auto-negotiation.

Syntax:

speed ten
hundred
auto

Ten The interface will operate at 10 Mbps.

Hundred
The interface will operate at 100 Mbps

Auto The interface will automatically select the speed (10 Mbps or 100 Mbps) depending on the link partner’s capability.

Accessing the 10/100-Mbps Interface Monitoring Process

To monitor information related to the 10/100-Mbps Ethernet Network Interface, access the interface monitoring process by doing the following:

1. At the OPCON prompt, enter **console**. For example:

   ```
   * console
   ```

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

   ```
   + configuration
   ```

   See “Configuration” on page 86 for sample output of the **configuration** command.
3. Enter the **network** command and the number of the Ethernet interface. In this example:

   ```
   + network 0
   ETH100>
   ```

   The 10/100-Mbps Ethernet monitoring prompt is displayed. You can now view information about the 10/100-Mbps Ethernet interface by entering monitoring commands.

10/100-Mbps Ethernet Interface Monitoring Commands

This section summarizes the 10/100-Mbps Ethernet monitoring commands. Enter commands at the ETH100> prompt. Table 30 lists the monitoring commands.

Table 30. Ethernet Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Chapter 17. Using Link Aggregation Groups

Link aggregation allows you to bundle together multiple physical Ethernet links into a Link Aggregation Group (LAG) that acts as a single physical interface.

LAGs can be used between two switches or between a switch and a server. Link aggregation ensures that frames transmitted over the LAG are not misordered or duplicated, allowing you to aggregate bandwidth between two destinations and providing the additional redundancy associated with having multiple paths.

Figure 18 shows a LAG between two IBM 8371s and Figure 19 shows a LAG between an IBM 8371 and a server. Each physical link in a LAG must be running at the same link speed and must be in full-duplex mode.

In Figure 18, all the unicast traffic from station A to station D will flow over a single link in the aggregation. However, the reverse traffic, from station D to station A, may flow over a different link. Similarly, the traffic between station B and station C may flow over other links in the aggregation. The source and destination MAC addresses determines the link over which the link unicast traffic will flow.

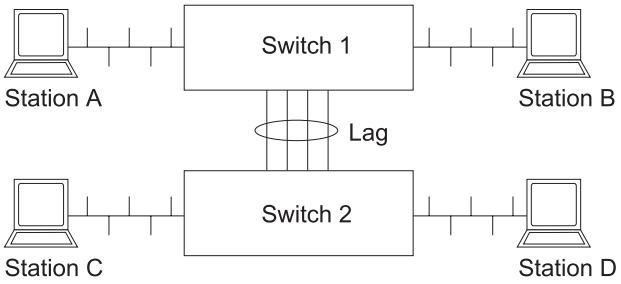

Figure 18. Link Aggregation Between two IBM 8371 Switches

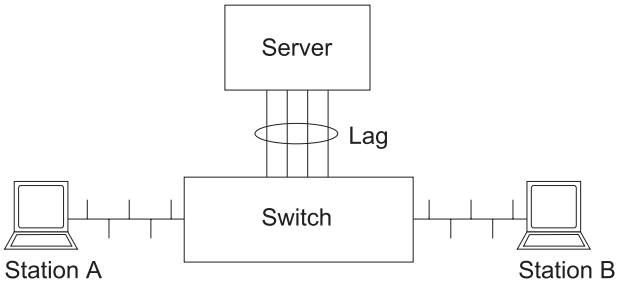

Figure 19. Link Aggregation Between an IBM 8371 and A Server

The IBM 8371 supports four LAG instances, with up to ten physical Ethernet links in each LAG. Routing, Self Learning IP, and MPOA client functions support the LAG function.

© Copyright IBM Corp. 1996, 1999

Chapter 18. Configuring and Monitoring Ethernet Link Aggregration Groups

This chapter describes how to configure and monitor Ethernet Link Aggregration Groups (LAGs).

- "Accessing the LAG Configuration Environment"
- "LAG Configuration Commands"
- "Accessing the LAG Monitoring Environment" on page 173
- "LAG Monitoring Commands" on page 173

Accessing the LAG Configuration Environment

Use the following procedure to access the LAG configuration commands. This process gives you access to the LAG *configuration* process.

1. At the OPCON prompt, enter **configuration**. For example:

```
* configuration
Config>
```

   After you enter the **configuration** command, the OPCON prompt (`Config>`) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the `Config>` prompt, enter **net *x***, where ***x*** is an integer value between 32 and 35 that specifies the network interface number of the LAG to be configured to get you to the `LAG Config` prompt.

   Example:

```
Config> net 32
LAG Config>
```

LAG Configuration Commands

Use the following commands to configure LAG. Enter these commands at the `LAG Config>` prompt.

Table 31. LAG Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds a link to a LAG.
Delete	Deletes a link from a LAG.
List	Displays the links currently configured on the LAG.
Set	Establishes or changes the configuration information for LAG.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

Add

Use the **add** command to add a link to a LAG.

Syntax:

add *interface#*

© Copyright IBM Corp. 1996, 1999

interface#

The specified physical interface must not be configured on the bridge and no IP or IPX protocol addresses can be defined on the interface.

Example:

```
LAG Config> add 1
```

Delete

Use the delete command to delete an interface from a LAG.

Syntax:

delete *interface#*

interface#

Deletes a link from a LAG.

Example:

```
LAG Config> delete 1
```

List

Use the **list** command to display LAG configuration information.

Syntax:

list

Lists all LAG-related configuration information.

Example:

```
Config>net 32
Link Aggregation Configuration
LAG config>add 1
LAG config>list
Link Aggregation Group:
Interface Number      : 32
Aggregation Type      : Sun Trunking 1.0/Fast EtherChannel
Flush Timer Value     : 1000ms
Interfaces in LAG     : 0  1
LAG config>exit
```

Set

Use the **set** command to establish or change the configuration information concerning LAG.

Syntax:

set flush-timer

flush-timer

Sets the duration of the flush timer in milliseconds.

Valid values: 100 - 4000

Default Value: 1000

Example:

```
LAG config>set flush-timer
Please enter new flush timer value [1000]? 4000
```

Accessing the LAG Monitoring Environment

Use the following procedure to access the LAG monitoring commands. This process gives you access to the LAG *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. For example:

```
* console
+
```

   After you enter the **console** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the + prompt, enter + **net *x***, where ***x*** is an integer value between 32 and 35 that specifies the network interface number of the LAG to be configured to get you to the `LAG Console` prompt.

   Example:

```
+ net 32
Lag Console>
```

LAG Monitoring Commands

This section summarizes the LAG monitoring commands.

Enter the LAG monitoring commands at the `LAG Console` prompt.

Table 32. LAG Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Dynamically adds a link to an aggregation group.
Delete	Dynamically removes a link from an aggregation group.
List	Displays current information about the aggregation.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **add** command to dynamically add a link to an aggregation group.

Note: The specified physical interface must not be a bridge port and must not have any IP or IPX addresses currently assigned to it.

Syntax:

add *interface#*

interface#
Dynamically adds a link to a LAG.

Example:

```
Lag Console> add 1
```

Delete

Use the **delete** command to dynamically remove a link from an aggregation group.

Syntax:

delete *interface#*

interface#
Dynamically removes a link from a LAG.

Example:

```
Lag Console> delete 1
```

List

Use the **list** command to display current information about the aggregation such as the LAG MAC address, the state of the virtual LAG interface, and the state of each member link.

Syntax:

list

Example:

```
+net 32
Lag Console>list
Aggregation Type : Sun Trunking 1.0/Fast EtherChannel
Flush Timer Value: 1000ms
LAG MAC: 0.4.ac.c4.d3.f0
LAG Net 32 is UP

Link Net #   Link State
----------   ----------
         1   Up
         0   Down
Lag Console>
```

If the links in the LAG have auto-negotiated to different speeds so that there is a speed mismatch, the following information will appear:

```
Lag Console>list
Aggregation Type : Sun Trunking 1.0/Fast EtherChannel
Flush Timer Value: 1000ms
LAG MAC: 0.4.ac.c4.d3.f0
LAG Net 32 is DOWN

LAG error: speed mismatch detected
Link Net #   Link State
----------   ----------
....
Lag Console>
```

Chapter 19. Using ATM

This chapter describes how to use the ATM interface. It includes the following sections:

- "ATM and LAN Emulation"
- "How to Enter Addresses"

ATM and LAN Emulation

LAN emulation provides support for 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, 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

   ```
   A1FF01020304
       or
   A1-FF-01-02-03-04
       or
   A1.FF.01.02.03.04
       or
   39.84.0F.00.00.00.00.00.00.00.00.00.03.10.00.5A.00.DE.AD.08
       or
   A1:FF:01:02:03:04
       or even
   A1-FF.01:0203:04
   ```

   Each type of address requires a different number of hexadecimal characters:

   ATM 40
   MAC 12
   ESI 12

   This information applies to addresses entered for LAN emulation and MPOA.

© Copyright IBM Corp. 1996, 1999

Chapter 20. Configuring and Monitoring ATM

This chapter describe the ATM interface configuration and operational commands. It includes the following sections:

Accessing the ATM Interface Configuration Process

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 "What is the OPCON Process?" on page 41.) 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 device is currently configured.
3. Enter the **network** command and the number of the ATM interface you want to configure. For example:

   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 33. ATM Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
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

© Copyright IBM Corp. 1996, 1999

Table 33. ATM Configuration Command Summary (continued)

Command	Function
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 21. Using LAN Emulation Clients" on page 189. • 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).
Assign-lec	Assigns a specified LEC to an ATM interface.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

ATM Interface Configuration Commands

This section summarizes and then explains the commands for configuring a specific ATM interface.

Enter the commands at the `ATM INTERFACE>` prompt.

Table 34. ATM INTERFACE Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
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 179.
Remove	Removes an ESI.
Set	Sets parameters of the ATM network.
Disable	Disables an ESI.
Enable	Enables an ESI.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

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 an 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

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 180.

Example: `list con`

```
                        ATM Configuration

Interface (net) number =     36
Maximum VCC data rate Mbps   =      155
Maximum frame size    = 1664
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
ATM Network ID =    0
```

esi 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”.

Syntax:

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 device 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-callers
max-calls
max-config-selectors
max-data-rate
max-frame
max-mp-parties
network-id
trace
uni-version

max-callers
Sets the maximum number of entities on the device that use the ATM interface. Each LEC 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-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-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 relative sizes of the selector range can be modified to conform to the types and numbers of ATM users on the device.

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-data-rate *speed*

Sets the default and upper bound for VCC traffic parameters of most LANE 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 device. The units for this parameter are Mbps.

Valid Values:

25
100
155

Default Value:

155

Example:

```
ATM INTERFACE> set speed 155
```

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 device entities using the ATM interface must use a maximum frame size less than or equal to the max-frame-size of the ATM interface.

Valid Values:

An integer in the range 512 - 31000

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 device. This parameter affects system memory allocation. Increasing this value is necessary if the device 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
```

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

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, off

Default Value:

off

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. If the UNI versions are configured on the ATM switch and ATM device interface to a specific version (not AUTO-DETECT), the UNI versions must match.

If the UNI version is configured as AUTO, the ATM device attempts to learn the UNI version to use from the switch.

In UNI AUTO-DETECT mode, if the switch does not respond to the query for UNI version, the default is UNI 3.0. If the switch responds with a value other than UNI 3.0 or UNI 3.1, the default is UNI 3.1.

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
```

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 device restart.

Syntax: disable esi *esi-address*

esi *esi-address*

Address of End System Identifiers.

Valid Values:

Any 12 hexadecimal digits

Default Value:

none

Example: `disable esi`

```
ATM INTERFACE> disable esi 00:00:00:00:00:09
```

Accessing the ATM Monitoring Process

Use the following procedure to access the ATM monitoring 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 "What is the OPCON Process?" on page 41.) For example:

   ```
   * talk 5
   +
   ```

   The GWCON prompt (+) is displayed on the console. If the prompt does not appear when you first enter the console, 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 36
   ATM+
   ```

   The ATM monitoring prompt (ATM+) is displayed.

ATM Monitoring Commands

This section summarizes the ATM monitoring commands for monitoring ATM interfaces. Enter the commands at the ATM+ prompt.

Table 35. ATM monitoring command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Interface	Displays the ATM Interface+ prompt from which you can monitor the ATM Interface, as described in "ATM Interface Monitoring Commands (ATM INTERFACE+ Prompt)".
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

Interface

Displays the ATM Interface+ prompt, described in "ATM Interface Monitoring Commands (ATM INTERFACE+ Prompt)".

Syntax:

interface

ATM Interface Monitoring 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 36. ATM INTERFACE monitoring command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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 you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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:

```
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:

```
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 device. 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:

P-P point to point VCC

P-MP point to multipoint VCC

ILMI Interim Local Management Interface VCC

SAAL signalling VCC

reserved-bandwidth
Lists the reserved bandwidth on the ATM Interface.

Example:

```
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 147.

Syntax:

trace list
on
off

list Displays the current packet tracing options on the ATM interface.

Example:

```
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:

```
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:

```
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:

```
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.

Assign-lec Configuration Command

Use the **assign-lec** command to assign a specified LEC to the ATM interface.

Syntax:

assign-lec

Select LEC to assign
Specifies the interface number of the LEC to be assigned to the ATM interface.

Note: The ATM interface is selected using the **net x** at the `Config>` prompt, where *x* is the physical ATM interface number.

Assign-lec Monitoring Command

Use the **assign-lec** command to dynamically assign a specified LEC to the ATM interface.

Syntax:

assign-lec

Select LEC to assign
Specifies the interface number of the LEC to be assigned to the ATM interface.

Note: The ATM interface is selected using the **net x** at the + prompt, where *x* is the physical ATM interface number.

Chapter 21. Using LAN Emulation Clients

This chapter describes LAN Emulation Clients (LECs). It includes the following sections:

- "LAN Emulation Client Overview"

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" on page 191
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.

© Copyright IBM Corp. 1996, 1999

Chapter 22. Configuring and Monitoring LAN Emulation Clients

This chapter describes how to configure LAN Emulation Clients (LECs). It includes the following sections:

Configuring LAN Emulation Clients

This section explains the commands for viewing, changing, and using the set of LE Clients on a particular ATM interface.

Enter the commands at the `LE Client Config>` prompt under the `ATM Config>` prompt, as described in "ATM Configuration Commands" on page 177.

To get to the `LE Client Config>` prompt, enter **le-c** at the ATM Config> prompt as described in "ATM Configuration Commands" on page 177.

Table 37. LAN EMULATION Client Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Config	Gets you to the `LEC Config>` prompt, from which you can configure a specific LAN Emulation Client as described in: • "Configuring an ATM Forum-Compliant LE Client" on page 192
List	Lists the LECs
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

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. Refer to "Configuring an ATM Forum-Compliant LE Client" on page 192.

Syntax:

config interface#

interface#

An integer number assigned by the device when the LEC was added to the configuration. Use the **list** command to determine the interface number assigned to the LEC.

Example:

```
LE Client Config> config 40
```

© Copyright IBM Corp. 1996, 1999

List

Use the **list** command to list the LAN emulation clients.

Syntax:

list

Example:

```
LE Client Config> list
                  ATM Emulated LANs
-----------------------------------------------------
  ATM interface number = 36
  LEC interface number = 40
  Emulated LAN type    = Ethernet Forum Compliant
  Emulated LAN name    =
```

Configuring an ATM Forum-Compliant LE Client

Use this process to access the appropriate `LEC Config>` prompt.:

1. Use the **config** command at the `LE Client Config>` prompt to access the appropriate LEC interface number, or use the **network** configuration command with the appropriate LEC interface number.
2. Enter the appropriate commands at the `Ethernet Forum Compliant LEC Config>` prompt.

This section explains the commands for configuring an ATM Forum-compliant LAN Emulation Client.

Table 38. LAN Emulation Client Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
ARP-Configuration	Allows you to configure the LE-ARP configuration for the ATM Forum-compliant client
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 `LEC QoS Config` prompt from which you can configure Quality of Service as described in “LE Client QoS Configuration Commands” on page 223.
Set	Sets the LAN Emulation Client parameters.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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 39. ATM LAN Emulation Client ARP Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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 you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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 (:).

Syntax:

add	mac
	route-descriptor

Example 1:

```
ARP config for LEC>add mac
MAC address of LE ARP Entry []? 123456789098
ATM address in 00.00.00.00.00.00:... form []? 390f000000000000000000000000123456789098
Destination Type -  REMOTE or LOCAL [Remote]?
```

Example 2:

```
ARP config for LEC>add route 12.34
ATM address in 00.00.00.00.00.00:... form []? 390f000000000000000000001234567890988888
ARP config for LEC>
```

Config

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 40. ATM LAN Emulation Client ARP Config Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Set	Sets QoS parameter values.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Set:

Use the **Set** command to configure the permanent ARP cache entry QoS parameters for the ATM forum-specific LAN Emulation Client.

Syntax:

set max-reserved-bandwidth
traffic-type
peak-cell-rate
sustained-cell-rate
qos-class
max-burst-size

Example:

```
ARP entry 'identifier' config> set ?
 MAX-RESERVED-BANDWIDTH
 TRAFFIC-TYPE
 PEAK-CELL-RATE
 SUSTAINED-CELL-RATE
 QOS-CLASS
 MAX-BURST-SIZE
```

See “Chapter 23. Configuring and Monitoring Quality of Service (QoS)” on page 217 for detailed information about the QoS parameters.

List

Use the **list** command to display information about ARP configuration.

Remove

Use the **remove** 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.

Syntax:

remove *arp-entry-number*

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-802.3.

Syntax:

IP-encapsulation Ethernet

IEEE-802.3

List

Use the **list** command to list the LE client configuration.

Syntax:

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 223.

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
bus-connect-retries
conn-completion-time
control-timeout
data-direct-timeout
data-direct-vcc-mode
elan-name
esi-address
flush-timeout
forward-delay
forward-disconnect-timeout
frame-size
initial-control-timeout
lecs-atm-address
les-atm-address

mac-address

multicast-send-avg

multicast-send-peak

multicast-send-type

multiplier-control-timeout

path-switch-delay

reconfig-delay-min

reconfig-delay-max

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:

```
LEC Config> set arp-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 65535.

Default Value:

5000

Example:

```
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:

```
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:

```
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 200.

Default Value:

NO

Example:

```
LEC Config> set 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:

```
LEC Config> set best-effort-peakrate 24000
```

bus-connect-retries

This parameter sets the maximum number of times that the LEC will attempt to reconnect to the BUS before returning to the initial state.

Valid Values:

0 - 2

Default Value:

1

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

Example:

```
LEC Config> set connection-completion-time 5
```

control-timeout

This parameter sets the maximum cumulative control timeout of a request.

A current timeout value is initialized to the value of ***initial-control-timeout***. If a response to a request is not received within the current timeout value, the current timeout is multiplied by the value of the ***multiplier-control-timeout*** and the request is reissued. Each time the current timeout value expires, this process is repeated until the current timeout value exceeds the value of ***control-timeout***.

Valid Values:

An integer number of seconds in the range of 10 to 300.

Default Value:

30

Example:

```
LEC Config> set control-timeout 100
```

data-direct-timeout

Specifies the timeout value for the data direct VCC. This parameter limits the time the Data Direct VCCs are left up without the LEC having a connection to the LES/BUS. If the LEC rejoins a LES/BUS before the timer expires, the time is stopped.

Valid Values:

10 - 300 seconds

Default Value:
30

data-direct-vcc-mode
Specifies whether persistent Data Direct VCC mode is enabled or disabled. When the Data Direct VCC mode is enabled, if the LEC loses its connection to the LES/BUS, the Data Direct VCCs are not dropped and the reconnect timeout timer is started. The LEC will continue to try to reconnect to the LES/BUS. If the LEC cannot reconnect to the LES/BUS before the **data-direct-timeout** expires, all Data Direct VCCs will be disconnected.

Valid Values:
yes or no

Default Value:
no

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:

```
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:
Burned-in ESI

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. During recovery, any queued frames are dropped and a new flush request is sent.

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.

Valid Values:
An integer number of seconds in the range of 1 to 4.

Default Value:
4

Example:

```
LEC Config> set flush-timeout 3
```

forward-delay
Sets the forward delay. Entries in the LE ARP cache must be periodically re-verified. 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 re-verification traffic.

Valid Values:
An integer number of seconds in the range of 4 to 30.

Default Value:
15

Example:

```
LEC Config> set forward-delay 10
```

forward-disconnect-timeout
This parameter sets the amount of time that a LEC will wait after losing its last Multicast Forward VCC from the BUS before returning to the initial state. This delay permits the BUS to attempt to reconnect to the client without returning to the initial state.

Valid Values:
10 - 300 seconds

Default Value:
60

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 181.

Valid Values:
1516

Default Value:
If the ELAN type is Ethernet, the default is 1516.

Example:

```
LEC Config> set frame-size 4544
```

initial-control-timeout
This parameter sets the value of the initial control timeout used in the control timeout algorithm described in 198.

Valid Values:
1 - 10

Default Value:
5

Example:

```
LEC Config> set initial-control-timeout 10
```

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:

```
LEC Config> set lecs-atm-address
39.84.0F.00.00.00.00.00.00.00.00.00.01.10.00.5A.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 auto-config** command on page 197.

- If auto-config is YES, the les-atm-address is not configurable.
- If 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:

```
LEC Config> set les-atm-address
      39.84.0F.00.00.00.00.00.00.00.00.00.01.10.00.5A.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:

```
LEC Config> set mac-address
  Use adapter address for MAC? [No]
  MAC address []: 10.00.5a.00.00.01
```

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 be less than or equal to 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:

```
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 be less than or equal to 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:

```
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:

```
LEC Config> set multicast-send-type best-effort
```

multiplier-control-timeout
This parameter sets the value of the control timeout multiplier used in the control timeout algorithm described on page 198.

Valid Values:
2 - 5

Default Value:
2

Example:

```
LEC Config> set multiplier-control-timeout 5
```

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:

```
LEC Config> set path-switch-delay 5
```

reconfig-delay-min
This parameter sets the minimum delay time when LEC returns to the initial state. This value must be ≤ ***reconfig-delay-max***.

Valid Values:
1 - the value of ***reconfig-delay-max***

Default Value:
1

Example:

```
LEC Config> set reconfig-delay-min  5
```

reconfig-delay-max
This parameter sets the maximum delay time when LEC returns to the initial state. This value must be ≥ ***reconfig-delay-min***.

Valid Values:
1 - 10

Default Value:
5

Example:

```
LEC Config> set reconfig-delay-max 9
```

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. If no ARP response is received after the specified number of retries, then the entry is purged from the LE ARP cache.

Valid Values:

0, 1, or 2

Default Value:

1

Example:

```
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 device. 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:

```
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:

Yes or No

Default Value:

No

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

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:

```
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.

Valid Values: 0 to 31536000 seconds (1 year).

Default Value: 1200

Note: This parameter is meaningful only for SVC connections.

Example:

```
LEC Config> set vcc-timeout 1000
```

Accessing the LEC Monitoring Environment

Use the following procedure to access the LEC monitoring 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 “What is the OPCON Process?” on page 41.) 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 device is currently configured, and enter the *interface number* for the LEC you wish to monitor. For example:

   ```
   + network ?
    1  : 1-port 10/100 Ethernet
    2  : 1-port 10/100 Ethernet
    3  : 1-port 10/100 Ethernet
   .
   .
   .
   30  : 1-port 10/100 Ethernet
   31  : 1-port 10/100 Ethernet
   36  : ATM
   37  : ATM
   38  : ATM
   39  : ATM
   40  : ATM Ethernet LAN Emulation: ELAN1
   41  : ATM Ethernet LAN Emulation: ELAN2
   .
   .
   .
     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 Monitoring Commands

This section summarizes and then explains the LEC monitoring commands. You can access LEC monitoring commands at the LEC+ prompt. Table 41 shows the commands.

Table 41. LE Client Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
List	Lists: • LEC Address Resolution Table (ARP) • LEC configuration • Data Direct VCC information • Group addresses • RIF information • LEC statistics • VCC table.
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 Monitoring Commands” on page 231.
Trace	Sets packet tracing on or off or sets a trace address or trace mask.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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
group
rif
statistics
vcc-table

arp Lists the LEC Address Resolution Table (entries in the ARP cache).

Example:

```
LEC+ list arp

        LEC Address Resolution (LE ARP Cache) Table

  Max Table Size          = 10
  Free Table Entries      = 10
  Current Mac Entries     = 0
  Current RD Entries      = 0
  Arp Aging Time          = 300
  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.

data Lists the LEC Data Direct VCC information.

Example:

```
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


 ------------------------------------------------------------------------------
```

group Lists the group addresses in use by the LEC.

statistics
Lists LEC statistics.

Example:

```
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
```

VCC table
Lists VCC table.

Example:

```
LEC+ list vcc
```

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:

```
LEC+ mib config

 lecConfigTable:
```

```
lecConfigMode                = Manual
lecConfigLanType             = 802.3 - Ethernet
lecConfigMaxDataFrameSize    = 1516
lecConfigLanName             =
lecConfigLesAtmAddress       = 39.84.0F.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
lecInitialControlTimeout       = 5
lecControlTimeoutMultiplier= 2
lecConfigV2Capable            = TRUE
lecForwardDisconnectTimeout   = 60
lecMinReconfigDelay          = 1
lecMaxReconfigDelay         = 5
lecMaxBusConnectRetries  = 1
ExplorerExclude                   = FALSE
Data direct VCC mode         = TRUE
Data direct timeout               = 20
```

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
Time to wait before sending a READY_QUERY

lecInitialControlTimeout
Specifies the maximum cumulative control timeout

lecControlTimeoutMultiplier
Specifies the control timeout multiplier

lecConfigV2Capable
Specifies whether the LEC is LANE version 2 capable

lecForwardDisconnectTimeout
Specifies the time period to wait after losing last Multicast Forward VCC

lecMinReconfigDelay
Specifies the minimum delay time the LEC waits in initial state

lecMaxReconfigDelay
Specifies the maximum delay time the LEC waits in initial state

lecMaxBusConnectRetries
Specifies the maximum BUS connect retries before returning to initial state

ExplorerExclude
Specifies whether to drop RIF explorer frames

Data Direct VCC Mode
Specifies the persistent Data Direct mode

Data Direct Timeout
Specifies the persistent Data Direct VCC Timeout

mac Displays the LEC MAC ARP Table

rd Displays the LEC Route Descriptor Table

server Displays the LEC MIB Server VCC Tables

Example:

```
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:

```
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:

```
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 ?
```

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 Monitoring Commands” on page 231.

Syntax:

qos-information

Trace

Use the **trace** command to turn packet tracing on or off on the LEC. See “Chapter 13. Configuring and Monitoring the Event Logging System (ELS)” on page 113 for more information.

Use the **trace mac-address** command to limit the data traced. A packet will only be traced if its destination or source MAC address logically ANDed with the trace MAC mask equals the trace MAC address logically ANDed with the trace MAC mask.

Syntax:

trace

Part 3. Features

© Copyright IBM Corp. 1996, 1999

Chapter 23. Configuring and Monitoring Quality of Service (QoS)

This chapter describes Quality of Service (QoS) configuration and operational commands for LAN and ELAN interfaces in the device. It contains the following sections:

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

- Using QoS for the LE Client, ATM Interface, or Emulated LAN provides the following benefits 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 Emulated LAN can be configured with QoS if the user wishes to provide QoS for all traffic in that ELAN. For example, an ELAN carrying SNA traffic can be given priority by configuring QoS parameters for that ELAN.

© Copyright IBM Corp. 1996, 1999

– An ATM Interface can be configured with QoS if a user wants 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 *accept-qos-parms-from-lecs* parameter can be configured only for an LE Client.

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:
no

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 40
Ethernet Forum Compliant LEC Config> qos-configuration
elan-x 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 36
ATM Config> interface
ATM Interface Config> qos-configuration
ATM-I/F 0 QoS>
```

Quality of Service Commands

This section summarizes the QoS configuration commands. Use the following commands to configure Quality of Service. Enter the commands from the QoS Config> prompt.

Table 42. Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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 43. LE Client Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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 you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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 = 36,  LEC interface number = 40)

  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 accept-qos-parms-from-lecs
all-default-values
max-burst-size
max-reserved-bandwidth
negotiate-qos
peak-cell-rate
qos-class
sustained-cell-rate
traffic-type
validate-pcr-of-best-effort-vccs

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 222 for a more detailed description of this parameter.

Valid Values:

yes, no

Default Value:

yes

Example:

```
LEC QoS Config> se acc y
LEC QoS Config>
```

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 = 36,  LEC interface number = 40)

     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-burst-size

Sets the desired maximum burst size in frames. See “Maximum Burst Size (max-burst-size)” on page 220 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>
```

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 218 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>
```

negotiate-qos

Use this option to enable/disable the LE Client's participation in QoS negotiation. See "Negotiate QoS (negotiate-qos)" on page 222 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>
```

peak-cell-rate

Sets the desired peak cell rate for Data Direct. See "Peak Cell Rate (peak-cell-rate)" on page 219 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>
```

qos-class

Sets the desired QoS Class for Data Direct VCCs. See "QoS Class (qos-class)" on page 220 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>
```

sustained-cell-rate

Sets the desired sustained cell rate for Data Direct VCCs. See "Sustained Cell Rate (sustained-cell-rate)" on page 219 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>
```

traffic-type

Sets the desired traffic for Data Direct VCCs. See “Traffic Type (traffic-type)” on page 219 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>
```

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 221 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>
```

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>
```

ATM Interface QoS Configuration Commands

Table 44. LE Client Quality of Service (QoS) Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
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 you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

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-burst-size
max-reserved-bandwidth
peak-cell-rate
qos-class
sustained-cell-rate
traffic-type

max-burst-size
Sets the desired maximum burst size in frames. See “Maximum Burst Size (max-burst-size)” on page 220 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>
```

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 218 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>
```

peak-cell-rate

Sets the desired peak cell rate for Data Direct VCCs. See “Peak Cell Rate (peak-cell-rate)” on page 219 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>
```

qos-class

Sets the desired QoS Class for Data Direct VCCs. See “QoS Class (qos-class)” on page 220 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>
```

sustained-cell-rate

Sets the desired sustained cell rate for Data Direct VCCs. See “Sustained Cell Rate (sustained-cell-rate)” on page 219 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>
```

traffic-type

Sets the desired traffic for Data Direct VCCs. See “Traffic Type (traffic-type)” on page 219 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>
```

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>
```

Accessing the QoS Monitoring Commands

Use the **feature** command from the GWCON process to access the Quality of Service monitoring commands. Enter the **feature** followed by the feature number (6) or short name (QoS). For example:

```
+feature qos
Quality of Service (QoS) - User Monitoring
QoS+
```

Once you access the QoS monitoring prompt, you can select the monitoring of a particular LE Client. To return to the GWCON prompt at any time, enter the exit command at the QoS monitoring prompt.

Alternatively, you can access the QoS Monitoring 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 monitoring prompt enter **qos-information**.

   Example:

```
+network 40
ATM Emulated LAN Monitoring
LEC+qos information
LE Client QoS Monitoring
LEC 40 QoS+
```

Quality of Service Monitoring Commands

This section summarizes the QoS monitoring commands. Enter these commands at the QoS+ prompt.

Table 45. Quality of Service (QoS) Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
le-client	Gets you to the `LE Client QoS console +` prompt for the selected LE client.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

LE Client QoS Monitoring Commands

This section summarizes the LE Client QoS monitoring commands. Enter the commands from the `LEC num QoS+` prompt.

Table 46. LE Client QoS Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
List	Lists the current LE Client QoS information. Options include: configuration parameters, TLVs, VCCs, and statistics.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

List

Use the **list** command to list the QoS related information of this LE Client.

Syntax:

list configuration-parameters
data-direct-VCCs (Detailed Information)
statistics
tlv-information
vcc-information

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 218.
- If the set of the first six QoS parameters described in “QoS Configuration Parameters” on page 218 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+
```

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-EFFORT 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+
```

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+
```

Configuring Quality of Service (QoS)

Chapter 24. Self Learning IP

Self Learning IP is a feature that enables a LAN switch that is inserted between routers and their associated LANs to dynamically determine IP routing information. Once the LAN switch determines routing information, it assumes responsibility for routing all local IP unicast data and transparently passes data it cannot route to the attached router.

Self Learning IP works by snooping on the contents of ARP packets, identifying adjacent routers and building an IP forwarding table. The forwarding table includes the following information:

- IP address of the station
- MAC address of the station
- Interface through which the station can be accessed
- Type of LAN encapsulation used by the station
- MAC address of the station's default router
- Timeout value indicating when the IP forwarding table entry is to be aged-out

The Self Learning IP function does not rely on any routing protocol, so it works seamlessly in networks that use RIP, OSPF, or IGRP.

Note: The Self Learning IP and routing functions are mutually exclusive since only one of these functions may be enabled at any given time.

Accessing the Self Learning IP Configuration Environment

Use the following procedure to access the Self Learning IP configuration process.

1. At the OPCON prompt, enter **talk 6**. For example:

```
* talk 6
Config>
```

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

2. At the CONFIG prompt, enter the **feature self** command to get to the `Self Learning IP Config>` prompt.

Self Learning IP Configuration Commands

Table 47. Self Learning IP Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Disable	Disables Self Learning IP.
Enable	Enables Self Learning IP in default mode.
One_to_one	Enables Self Learning IP in one-to-one mode.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

© Copyright IBM Corp. 1996, 1999

Disable

Use the **disable** command to disable Self Learning IP.

Syntax:

disable

Enable

Use the **enable** command to enable Self Learning IP in default mode.

Syntax:

enable

One-to-one

When this mode is enabled, the ports on the IBM 8371 are paired. Either of the ports of the dedicated pair may then be connected to a router port and the other port of the dedicated pair is connected to the router's LAN interface. In this mode, broadcast frames received at one port will be transmitted only on the other port of the pair.

To disable One-to-one mode, use the **enable** command to enable Self Learning IP in default mode.

Syntax:

one-to-one

Accessing the Self Learning IP Monitoring Environment

Use the following procedure to access the Self Learning IP monitoring process.

1. At the OPCON prompt, enter **talk 5**. For example:

```
* talk 5

CGW Operator Console

+
```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter the monitoring environment, press **Return** again.

2. At the + prompt, enter the **feature self learning ip** command to get to the `Self Learning IP Console>` prompt.

Self Learning IP Monitoring Commands

Table 48. Self Learning IP Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Disable	Dynamically disables Self Learning IP.
Enable	Dynamically enables Self Learning IP.

Table 48. Self Learning IP Monitoring Command Summary (continued)

Command	Function
Hosts	Views the discovered hosts.
Routers	Views the discovered routers.
State	Displays the state of Self Learning IP.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Disable

Use the **disable** command to dynamically disable Self Learning IP.

Syntax:

disable

Enable

Use the **enable** command to dynamically enable Self Learning IP.

Syntax:

enable

Hosts

Use the **hosts** command to display discovered hosts.

Syntax:

hosts

```
Self Learning IP Console> hosts


Host IP Addr. | Host MAC Address  | IF | EN |    Router MAC     | Ttl
--------------+------------------+----+----+-------------------+-----
1.1.1.10      | 00:00:00:00:6E:00 | 00 | DX | 00:00:00:00:82:00 | 56 *
2.2.2.20      | 00:00:00:00:78:00 | 01 | SN | 00:00:00:00:82:01 | 56 *
```

Host IP Addr
Specifies the Host IP address.

Host MAC Address
Specifies the Host MAC address

IF Specifies the interface number on which the host has been found.

EN Specifies the encapsulation type. The encapsulation types are SN-LLC/SNAP and DX-DIX.

Router MAC
Specifies the MAC address of this host’s router.

Ttl Specifies the Time to Live for this host entry in this table.

***** Indicates that this entry has an active shortcut.

Routers

Use the **routers** command to display discovered routers.

Syntax:

routers

```
Self Learning IP Console> routers


IF │ Router MAC Address    │ State      │ Ttl
---+----------------------+------------+-----
01 │ 00:00:00:00:78:00     │RESOLVED    │ 198
00 │ 00:00:00:00:6E:00     │RESOLVED    │ 199
03 │ 00:00:00:00:82:01     │RESOLVED    │ 199
02 │ 00:00:00:00:82:00     │RESOLVED    │ 199
```

IF Specifies the interface to which the router is attached.

Router MAC
Specifies the router MAC address.

State Specifies the interface state.

Ttl Specifies the Time to Live for this router entry in this table.

State

Use the **state** command to display the current state of Self Learning IP.

Syntax:

state

Example:

```
Self Learning IP Console> state
Self Learning IP Enabled Flag is ON
Current Microcode load           Self Learning IP
Self Learning IP Flag in SRAM is ON
Self Learning IP Operation Mode  Default
Number of Routers found          4
Number of Hosts found            10
Memory blocks allocated          32
Memory blocks freed              12
Self Learning IP Console>
```

Chapter 25. Remote Network Monitoring

Remote Network Monitoring (RMON)is a standardized traffic monitor based upon SNMP. For details on RMON MIB implementation, refer to the README files at the FTP site described in “SNMP Management” on page 533.

Accessing the RMON Configuration Environment

Use the following procedure to access the RMON *configuration* process.

1. At the OPCON prompt, enter **talk 6**. For example:

```
* talk 6
Config>
```

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

2. At the CONFIG prompt, enter the **feature rmon** command to get to the `RMON Config>` prompt.

RMON Configuration Commands

Table 49. RMON Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Enable	Enables RMON, on reload. RMON defaults to *enabled*.
Disable	Disables RMON, on reload.
List	Displays RMON’s next state upon reload.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Disable

Use the **disable** command to disable RMON upon the next reload.

Syntax:

disable

Enable

Use the **enable** command to enable RMON upon the next reload.

Syntax:

enable

List

Use the **list** command to display RMON’s next state upon reload.

Syntax:

© Copyright IBM Corp. 1996, 1999

list

```
RMON Config> list

RMON                        = Enabled
```

Accessing the RMON Monitoring Environment

Use the following procedure to access the RMON *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. For example:

   ```
   * talk 5
   +
   ```

   After you enter the **talk 5** command, the CONFIG prompt (+) displays on the terminal. If the prompt does not appear, press **Return** again.
2. At the + prompt, enter the **feature rmon** command to get to the `RMON Console>` prompt.

RMON Monitoring Commands

Table 50. RMON Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Enable	Dynamically enables RMON.
Disable	Dynamically disables RMON.
Memstats	Displays collected memory statistics.
List	Displays current RMON status.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Disable

Use the **disable** command to dynamically disable RMON.

Syntax:

disable

Enable

Use the **enable** command to dynamically enable RMON.

Syntax:

enable

Memstats

Use the **memstats** command to display statistics.

Syntax:

memstats

```
RMON Config> memstats

RMON memory size in use               = 170368 Bytes
RMON memory requests made             = 176
```

RMON memory size in use
Specifies the amount of memory currently in use for RMON.

RMON memory requests made
Specifies the number of requests made for memory.

List

Use the **list** command to display the current state of RMON.

Syntax:

list

```
RMON Config> list

RMON                      = Enabled
```

Part 4. Protocols

© Copyright IBM Corp. 1996, 1999

Chapter 26. Bridging Methods

This chapter describes the methods of bridging supported by the adaptive source routing transparent (ASRT) bridge. Each section gives an overview of a specific technology and is followed by a description of the data frames supported by that technology. The chapter includes the following sections:

- "Transparent Bridging"

Transparent Bridging

The transparent bridge is also commonly known as a spanning tree bridge (STB). The term *transparent* refers to the fact that the bridge silently forwards non-local traffic to attached LANs in a way that is *transparent* or unseen to the user. End station applications do not know about the presence of the bridge. The bridge learns about the presence of end stations by listening to traffic passing by. From this listening process it builds a database of end station addresses attached to its LANs.

For each frame it receives, the bridge checks the frame's destination address against the ones in its database. If the frame's destination is an end station on the same LAN, the frame is not forwarded. If the destination is on another LAN, the frame is forwarded. If the destination address is not present in the database, the frame is forwarded to all the LANs that are connected to the bridge except the LAN from which it originated.

All transparent bridges use the spanning tree protocol and algorithm. The spanning tree algorithm produces and maintains a loop-free topology in a bridged network that might contain loops in its physical design. In a mesh topology where more than one bridge is connected between two LANs, *looping* occurs. In such cases, data packets bounce back and forth between two LANs on parallel bridges. This creates a redundancy in data traffic and produces the phenomenon known as looping.

When looping occurs, you must configure the local and/or remote LAN to remove the physical loop. With spanning tree, a self-configuring algorithm allows a bridge to be added anywhere in the LAN without creating loops. When the new bridge is added, the spanning tree protocol automatically reconfigures all bridges on the LAN into a single loop-free *spanning tree*.

A spanning tree never has more than one active data route between two end stations, thus eliminating data loops. For each bridge, the algorithm determines which bridge ports can forward data and which ones must be blocked to form a loop-free topology. The features that spanning tree provides include:

- *Loop detection*. Detects and eliminates physical data link loops in extended LAN configurations.
- *Automatic backup of data paths*. The bridges connecting to the redundant paths enter backup mode automatically. When a primary bridge fails, a backup bridge becomes active.
- *User configurability*. Lets you tailor your network topology. Sometimes the default settings do not produce the desired network topology. You can adjust the bridge priority, port priority, and path cost parameters to shape the spanning tree to your network topology.

© Copyright IBM Corp. 1996, 1999

Bridging Methods

- *Seamless interoperability*. Allows LAN interoperability without configuration limitations caused by diverse communications environments.
- *Bridging of non-routing protocols*. Provides cost-effective bridging of non-routing protocols.

Network Requirements

Transparent Bridge implements a spanning tree bridge that conforms to the IEEE 802.1D standard. All transparent bridges on the network must be 802.1D spanning tree bridges. This spanning tree protocol is not compatible with bridges implementing the proprietary Digital Equipment Corporation spanning tree protocol used in some older bridges.

Transparent Bridge Operation

In a mesh topology where more than one bridge is connected between two LANs, a looping phenomenon can occur where two LANs bounce packets back and forth over parallel bridges. A loop is a condition where multiple data paths exist between two LANs. The spanning tree protocol operating automatically eliminates loops by blocking redundant paths.

During startup, all participating bridges in the network exchange Hello bridge protocol data units (BPDUs) which provide configuration information about each bridge. BPDUs include information such as the bridge ID, root ID, and root path cost. This information helps the bridges to unanimously determine which bridge is the root bridge and which bridges are the designated bridges for LANs to which they are connected.

Of all the information exchanged in the HELLO messages, the following parameters are the most important for computing the spanning tree:

- *root bridge ID*. The root bridge ID is the bridge ID of the bridge. The root bridge is the designated bridge for all the LANs to which it is connected.
- *Root Path Cost*. The sum total of the designated path costs to the root via this bridge's root port. This information is transmitted by both the root bridge and the designated bridges to update all bridges on path information if the topology changes.
- *bridge ID*. A unique ID used by the spanning tree algorithm to determine the spanning tree. Each bridge in the network is assigned a unique bridge identifier.
- *port ID*. The ID of the port from which the current HELLO BPDU message was transmitted.

With this information available, the spanning tree begins to determine its shape and direction and then creates a logical path configuration. This process can be summarized as follows:

1. A root bridge for the network is selected by comparing the bridge IDs of each bridge in the network. The bridge with the lowest ID (that is, highest value) wins.
2. The spanning tree algorithm then selects a designated bridge for each LAN. If more than one bridge is connected to the same LAN, the bridge with the smallest path cost to the root is selected as the designated bridge. In the case of duplicate path costs, the bridge with the lowest bridge ID is selected as the designated bridge.
3. The non-designated bridges on the LANs put each port that has not been selected as a root port into a BLOCKED state. In the BLOCKED state, a bridge

still listens to Hello BPDUs so that it can act on any changes that are made in the network (for example, designated bridge fails) and change its state from BLOCKED to FORWARDING (that is, it will be forwarding data).

Through this process, the spanning tree algorithm reduces a bridged LAN network of arbitrary topology into a single spanning tree. With the spanning tree, there is never more than one active data path between any two end stations, thus eliminating data loops. For each bridge on the network, the spanning tree determines which bridge ports to block from forming loops.

This new configuration is bounded by a time factor. If a designated bridge fails or is physically removed, other bridges on the LAN detect the situation when they do not receive Hello BPDUs within the time period set by the bridge maximum age time. This event triggers a new configuration process where another bridge is selected as the designated bridge. A new configuration is also created if the root bridge fails.

Shaping the Spanning Tree

When the spanning tree uses its default settings the spanning tree algorithm generally provides acceptable results. The algorithm, however, may sometimes produce a spanning tree with poor network performance. In this case you can adjust the bridge priority, port priority, and path cost to shape the spanning tree to meet your network performance expectations. The following examples explain how this is done.

Figure 20 on page 250 shows three LANs networked using three bridges. Each bridge is using default bridge priority settings for its spanning tree configuration. In this case, the bridge with the lowest physical address is chosen as the root bridge because the bridge priority of each bridge is the same. In this example, this is Bridge 2.

The newly configured spanning tree stays intact due to the repeated transmissions of Hello BPDUs from the root bridge at a preset interval (bridge hello time). Through this process, designated bridges are updated with all configuration information. The designated bridges then regenerate the information from the Hello BPDUs and distribute it to the LANs for which they are designated bridges.

Table 51. Spanning Tree Default Values

Bridge 1	**Bridge 2**	**Bridge 3**
Bridge Priority: 32768 Address: 00:00:90:00:00:10	Bridge Priority: 32768 Address: 00:00:90:00:00:01	Bridge Priority: 32768 Address: 00:00:90:00:00:05
Port 1 Priority: 128 Path Cost: 100	**Port 1** Priority: 128 Path Cost: 100	**Port 1** Priority: 128 Path Cost: 100
Port 2 Priority: 128 Path Cost: 17857	**Port 2** Priority: 128 Path Cost: 17857	**Port 2** Priority: 128 Path Cost: 17857
Port 3 Priority: 128 Path Cost: 17857	**Port 3** Priority: 128 Path Cost: 17857	**Port 3** Priority: 128 Path Cost: 17857

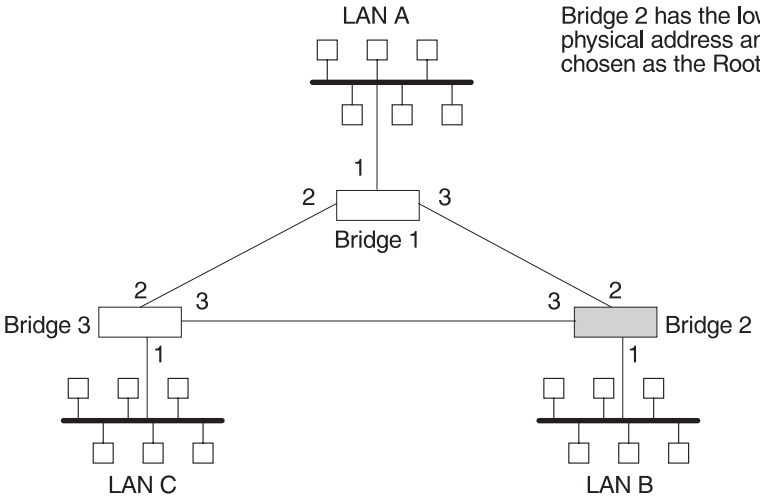

Figure 20. Networked LANs Before Spanning Tree

The spanning tree algorithm designates the port connecting Bridge 1 to Bridge 3 (port 2) as a backup port and blocks it from forwarding frames that would cause a loop condition. The spanning tree created by the algorithm using the default values in Table 51 on page 249 is shown in Figure 21 as the heavy lines connecting Bridge 1 to Bridge 2, and then Bridge 2 to Bridge 3. The root bridge is Bridge 2.

This spanning tree results in poor network performance because the workstations on LAN C can get to the file server on LAN A only indirectly through Bridge 2 rather than using the direct connection between Bridge 1 and Bridge 3.

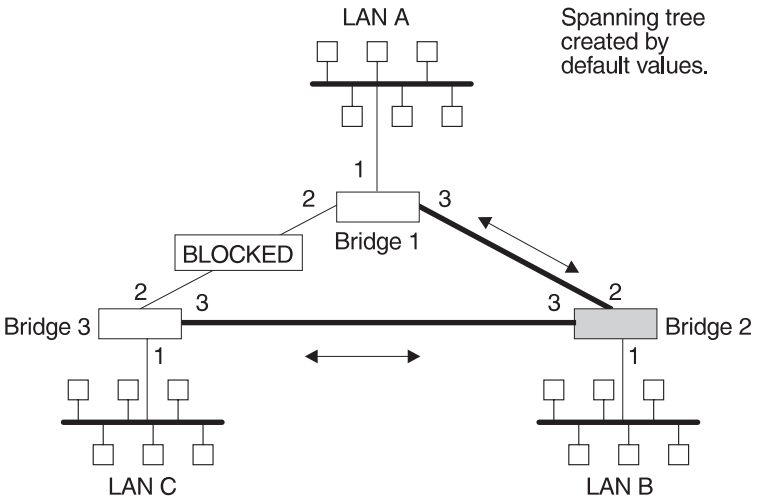

Figure 21. Spanning Tree Created With Default Values

Normally, this network uses the port between Bridge 2 and Bridge 3 infrequently. Therefore you can improve network performance by making Bridge 1 the root bridge of the spanning tree. You can do this by configuring Bridge 1 with the highest priority of 1000. The spanning tree that results from this modification is shown in Figure 22 on page 251 as the heavy lines connecting Bridge 1 to Bridge 3 and Bridge 1 to Bridge 2. The root bridge is now Bridge 1. The connection between Bridge 2 and Bridge 3 is now blocked and serves as a backup data path.

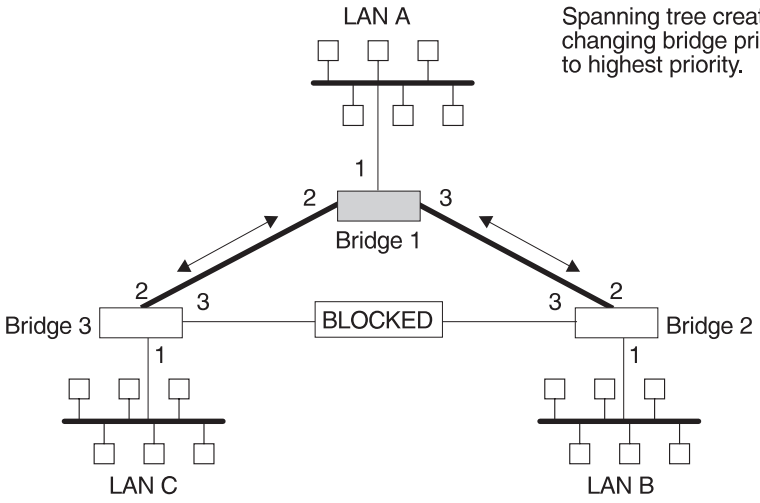

Figure 22. User-Adjusted Spanning Tree

Transparent Bridging and ATM

The ATM interface forwards transparent frames from Ethernet networks, provided bridging is enabled on the virtual channel connection (VCC).

Hello BPDUs are generated and transmitted for each LEC configured for transparent bridging. The spanning tree protocol causes ATM LECs that have not been designated as part of the active data path to be BLOCKED, thereby eliminating loops.

Transparent Bridge Terminology and Concepts

This section reviews the terms and concepts commonly used in transparent bridging.

Aging Time

The length of time (age) before a dynamic entry is removed from the filtering database when the port with the entry is in the forwarding state. If dynamic entries are not referenced by the aging time, they are deleted.

Bridge

A protocol-independent device that connects local area networks (LANs). These devices operate at the data link layer, storing and forwarding data packets between LANs.

Bridge Address

The least significant 6-octet part of the bridge identifier used by the spanning tree algorithm to identify a bridge on the network. The bridge address is set to the MAC address of the lowest-numbered port by default. You can override the default address by using the **set bridge** configuration command.

Bridge Hello Time

The bridge hello time specifies how often a bridge sends out Hello BPDUs (containing bridge configuration information) when it becomes the root bridge in the spanning tree. This value is useful only for the root bridge because it controls the hello time for all bridges in the spanning tree. Use the **set protocol bridge** command to set the bridge hello time.

Bridge Forward Delay

The amount of time a bridge port spends in the listening state as well as the learning state. The forward delay is the amount of time the bridge port listens in order to adjust the spanning tree topology. It is also the amount of time the bridge spends learning the source address of every packet that it receives while the spanning tree is configuring. This value is useful only for the root bridge because it controls the forward delay for all bridges in the spanning tree.

The root bridge conveys this value to all bridges. This time is set with the **set protocol bridge** command. The procedure for setting this parameter is discussed in the next chapter.

Bridge Identifier

A unique identifier that the spanning tree algorithm uses to determine the spanning tree. Each bridge in the network must have a unique bridge identifier.

The bridge identifier consists of two parts: a least-significant 6-octet bridge address and a most-significant 2-octet bridge priority. By default, the bridge address is set to the MAC address of the lowest-numbered port. You can override the default address with the **set bridge** configuration command.

Bridge Maximum Age

The amount of time that spanning tree protocol information is considered valid before the protocol discards the information and a topology changes. All the bridges in the spanning tree use this age to time out the received configuration information in their databases. This can cause a uniform timeout for every bridge in the spanning tree. Use the **set protocol bridge** command to set the bridge maximum age.

Bridge Priority

The most significant 2-octet part of the bridge identifier set by the **set protocol bridge** command. This value indicates the chances of each bridge becoming the root bridge of the network. In setting the bridge priority, the spanning tree algorithm chooses the bridge with the highest priority value to be the root bridge of the spanning tree. A bridge with the lowest numerical value has the highest priority value.

Designated Bridge

The bridge that claims to be the closest to the root bridge on a specific LAN. This closeness is measured according to the accumulated path cost to the root bridge.

Designated Port

The port ID of the designated bridge attached to the LAN.

Filtering and Permanent Databases

Databases that contain information about station addresses that belong to specific port numbers of ports connected to the LAN.

The filtering database is initialized with entries from the permanent database. These entries are permanent and survive power on/off or system resets. You can add or delete these entries through the spanning tree configuration commands. Entries in the permanent database are stored as static random access memory (SRAM) records, and the number of entries is limited by the size of SRAM.

Note: You can also add entries (static) by using the monitoring commands but these ***do not*** survive power on/off and system resets.

The filtering database also accumulates entries learned by the bridge (dynamic entries) which have an aging time associated with them. When entries are not referenced over a certain time period (age time), they are deleted. Static entries are ageless, so dynamic entries cannot overwrite them.

Entries in the filtering and permanent databases contain the following information:

- *Address*. The 6-byte MAC address of the entry
- *Port Map*. Specifies all port numbers associated with that entry
- *Type of Entry*. Specifies one of the following types:
  - Reserved Entries. Reserved by the IEEE 802.1d committee.
  - Registered Entries. Consist of unicast addresses belonging to communications hardware attached to the box or multicast addresses enabled by protocol forwarders.
  - Permanent Entries. Entered by the user in the configuration process. They survive power on/off and system resets.
  - Static Entries. Entered by the user in the monitoring process. They do not survive power on/off and system resets and are ageless.
  - Dynamic Entries. Dynamically learned by the bridge. They do not survive power on/off and system resets and have an associated age.
  - Free. Locations in database that are free to be filled by address entries.
- *Address Age (dynamic entries only)*. Resolution of time period at which address entries are ticked down before being discarded. You can set this value.

Make changes to the permanent database through the spanning tree configuration commands and make changes to the filtering database through the GWCON monitoring process.

Parallel Bridges

Two or more bridges connecting the same LANs.

Path Cost

Each port interface has an associated path cost which is the relative value of using this port to reach the root bridge in a bridged network. The spanning tree algorithm uses the path cost to compute a path that minimizes the cost from the root bridge to all other bridges in the network topology. The sum total of all the designated costs and the path cost of the root port is called the root path cost.

Port

The bridge's connection to each attached LAN or WAN. A bridge must have at least two ports to function as a bridge.

Port ID

A 2-octet port identifier. The most-significant octet represents the port priority and the least-significant octet represents the port number. Both port number and port priority are user-assignable. The port ID must be unique within the bridge.

Port Number

A user-assigned 1-octet part of the port ID whose value represents the attachment to the physical medium. A port number of zero is not allowed.

Port Priority

The second 1-octet part of the port ID. This value represents the priority of the port that the spanning tree algorithm uses in making comparisons for port selection and blocking decisions.

Resolution

The time factor by which dynamic entries are ticked down as they age within the database. The range is 1 to 60 seconds.

Root Bridge

The bridge selected as the *root* of the spanning tree because it possesses the highest priority bridge ID. This bridge is responsible for keeping the spanning tree intact by regularly emitting Hello BPDUs (containing bridge configuration information). The root bridge is the designated bridge for all the LANs to which it is connected.

Root Port

The port ID of a bridge's port that offers the lowest cost path to the root bridge.

Spanning Tree

A topology of bridges such that there is one and only one data route between any two end stations.

Transparent Bridging

This type of bridging involves a mechanism that is *transparent* to end stations applications. Transparent bridging interconnects local area network segments by bridges designated to forward data frames through a spanning tree algorithm.

Chapter 27. Bridging Features

This chapter describes bridging features that are available with the Adaptive Source Routing Transparent (ASRT) bridge. The chapter includes the following sections:

- "TCP/IP Host Services (Bridge-Only Management)"
- "Bridge-MIB Support"

TCP/IP Host Services (Bridge-Only Management)

The IBM 8371 also supports TCP/IP Host services, which let you configure and monitor a bridge . This option gives you the following capabilities:

- Management through SNMP
- Telnet server function
- Downloading and uploading of configurations through the TFTP protocol
- TFTP neighbor boot function
- IP diagnostic tools of ping and trace route
- Control of the device through SNMP sets and the telnet client

When viewed from the bridge's monitoring interface, TCP/IP Host Services is handled as a new protocol having its own configuration and monitoring prompts. These prompts are accessed via the **protocol** command in `talk 6` and `talk 5`.

Bridge-MIB Support

For Bridge Management via SNMP, the IBM 8371 supports the management information bases (MIBs) as specified by RFC 1493 and RFC 1525, ***except*** for the following MIBs:

- dot1dStaticTable
- dot1dTpFdbTable
- dot1dPortPairTable

Dynamic Protocol Filtering VLANs

Dynamic protocol filtering (DPF) VLANs are based on protocol and subnets, in addition to user-defined traffic types. For each configured vlan, the subset of bridge ports on which traffic for that vlan is received is the forwarding domain of that vlan. Dynamic protocol filtering (DPF) can partition the bridged network into:

- IP, IPX, and NetBIOS Protocol VLANs

  DPF monitors traffic on each bridge port and learns the location of traffic matching the configured protocols and subnets. Ports with matching inbound traffic are included in the forwarding domain of the corresponding VLAN.

- IP Multicast VLANs

  IP Multicast VLANs restrict IP multicast data to ports in the same IP multicast group and ports attached to multicast devices. IGMP Report frames are used to determine port inclusion in the forwarding domain for a particular IP Multicast VLAN. Also, ports receiving IGMP Query device frames are included in the forwarding domain of all IP Multicast VLANs to insure all IP multicast data is sent to the multicast devices.

The purpose of DPF is to limit the proliferation of frames that are normally forwarded over all active spanning tree ports. DPF dynamically activates filters

© Copyright IBM Corp. 1996, 1999

based upon the traffic on each bridge port. The bridged network can thus be dynamically partitioned into protocol-specific subnetworks.

DPF offers further benefits to increase performance, enhance security and facilitate moves and changes in the network.

For subnetted IP networks, DPF has an *IP-cut-through* facility that allows establishment of data-direct VCCs between IP workstations on different IP subnet VLANs. By enabling *IP-cut-through* and shortening the IP subnet mask in end-stations, the end-stations communicate directly with each other without involving an IP device. This significantly increases IP throughput in the network, reduces IP routing requirements, and isolates IP subnet broadcast traffic.

IP-cut-through can be enabled or disabled by an IP subnet or IP end-station. *IP-cut-through* can also be configured to allow cut-through in one direction but force a routed path in the reverse direction. This uni-directional cut-through can be used to force IP clients to go through an IP device for security but allow IP servers to “cut through” to the clients for maximum performance.

Since DPF automatically adjusts the forwarding domain of a VLAN based on traffic, it lets users move around the network without any changes to their configuration. This is especially useful for IP networks, because it eliminates the need for assigning new IP addresses when users move.

DPF is a bridging enhancement. All ports on the ASRT bridge environment must be the same type. VLANs can be configured for multiple IP subnets, multiple IPX networks, a single NetBIOS network, user-defined traffic types, and IP multicast groups.

Required Static Configurations

You must statically configure VLAN ports in the following situations:

- Ports with devices with low network utilization.

  Devices such as printers, servers or devices on a port could lose connectivity because of low network utilization. To prevent aging-out of a port that defines a VLAN to such a device, configure the port statically; specify **include** when prompted to configure the VLAN on the port. For example:
- A bridge port connected to IPX clients only.

  IPX clients do not know their network numbers. This prevents a VLAN from learning the association between the network number and the port number. Specify **include** when prompted to configure the VLAN for a bridge port connected to IPX clients only.

IP-Cut_Through Considerations

IP Cut-Through enables communication between stations on different IP subnets. IP Cut-Through is applicable in subnetted IP networks only. If stations are on different IP nets, then communication cannot be established between them and a device must be used to forward traffic between those stations.

To use IP Cut-Through, the subnet mask in end-stations (typically just servers) should be shortened. That is, a 255.255.255.255 subnet mask is shortened to 255.255.255.0 to imply a 3-byte subnet and a 255.255.0.0 subnet mask implies a 2-byte subnet. Shortening the subnet mask will cause the end-station to ARP for the

destination and establish communication to the destination (or intermediate LAN switch), maximizing network throughput. However, this configuration can produce the following side effects:

1. A large number of ARP entries can be created in end-stations with a shortened mask which in turn can increase their CPU utilization.

   If these end-stations are ATM-attached, the number of ATM connections (data-direct VCCs) will also increase.

   Therefore, the need for faster network throughput must be balanced against increased CPU utilization in the end-stations and increased VCC utilization in the ATM switches.
2. An end-station with a shortened mask could ARP for a destination that is not directly connected. For example, this can happen if the destination is on a different type of LAN or behind a device firewall. The only way to reach this destination is through a device but devices normally do not propagate ARPs between networks. This scenario can work only when the Proxy ARP function is enabled in the device. This will cause the device to respond to the ARP and subsequent traffic will be sent to the device.

Answering Yes to the **Enable IP-Cut-Through from this VLAN?** question will allow forwarding of IP traffic from devices on this VLAN to devices on other VLANs that have IP-Cut-Through reception enabled.

Auto-created IP Multicast VLANs

Unlike other VLANs, IP Multicast VLANs can be automatically created and configured without user involvement. If auto-creation of IP Multicast VLANs is enabled, then the receipt of an IGMP Report frame (indicating a station's membership in an IP multicast group) causes an IP Multicast VLAN to be created for the group address indicated in the frame. Thus, IP Multicast groups can be configured on stations in the network without the need for VLAN configuration in the MSS bridge.

Auto-creation is enabled if an IP Multicast VLAN exists for the all IP hosts address of 224.0.0.1 and is enabled. If not already present, this VLAN is created and enabled during box initialization. It contains the initial port configuration, aging time, and MAC Address tracking status that will be applied to each new IP Multicast VLAN that is automatically created. To turn off auto-creation of IP Multicast VLANs, disable the VLAN for the 224.0.0.1 group address.

No IP Multicast VLANs can be auto-created or manually configured for the reserved multicast groups whose address is between 224.0.0.0 and 224.0.0.255, inclusive. This prevents potential problems in filtering frames necessary to several protocols that use these addresses.

Chapter 28. Configuring and Monitoring Bridging

This chapter describes how to configure the adaptive source routing transparent (ASRT) bridge protocol and how to use the ASRT configuration commands. The chapter includes the following sections:

- "Accessing the ASRT Configuration Environment"
- "ASRT Configuration Commands"

Accessing the ASRT Configuration Environment

To access the ASRT configuration environment, enter the **protocol asrt** command at the `Config>` prompt:

```
Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>
```

ASRT Configuration Commands

The ASRT configuration commands allow you to specify network parameters for the ASRT bridge and its network interfaces. These commands also allow you to enable and configure the ATM interface features, and NetBIOS.

The device must be restarted for the new configuration to take effect.

Note: The ASRT configuration commands are not effective immediately. They remain pending until you reload the device.

Enter the ASRT configuration commands at the `ASRT config>` prompt. Access the commands as follows:

- Enter the configuration commands for dynamic protocol filtering (Virtual LANs) at the `VLAN config>` prompt. The VLAN prompt is accessed by entering the **vlans** command explained later in this chapter.

Table 52 shows the ASRT configuration commands.

Table 52. ASRT Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds station address entries to the permanent database.
Delete	Deletes station address entries.
Disable	Disables the following functions: • Bridging • Propagation of Spanning Tree Explorer Frames • Transparent (spanning tree) bridging function on a given port
Enable	Enables the following functions: • Bridging • Propagation of Spanning Tree Explorer Frames • Transparent (spanning tree) bridging function on a given port

© Copyright IBM Corp. 1996, 1999

Table 52. ASRT Configuration Command Summary (continued)

Command	Function
List	Displays information about the complete bridge configuration or about selected configuration parameters.
Set	Sets the following parameters: • Aging time for dynamic address entries • Bridge address • Maximum frame size • Spanning tree protocol bridge and port parameters • Filtering database size • Ethernet Preference
vlans	Allows the user to configure dynamic protocol filtering
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Response to ASRT Configuration Commands

The ASRT configuration (Talk 6) commands are not effective immediately. They remain pending until you issue the **reload** command.

Add

Use the **add** command to add the following information to your bridging configuration:
- Station address entries to the permanent database
- LAN/WAN ports

Syntax:

add address . . .

port . . .

address *addr-value*

Adds unique station address entries to the permanent database. These entries are copied into the filtering database as permanent entries when the bridge is restarted. The *addr-value* is the MAC address of the desired entry. It can be an individual address, multicast address, or broadcast address. You are also given the option to specify the outgoing forwarding port map for each incoming port. Permanent database entries are not destroyed by the power off/on process and are immune to the aging settings. Permanent entries cannot be replaced by dynamic entries.

Valid Values: X’0000 0000 0000’ to X’FFFF FFFF FFFF’

Default Value: none

The following sections present specific examples of how the **add address** command is used to manage address entries:

Adding an address

```
add address
Address (in 12-digit hex) []? 123456789013
    Exclude destination address from all ports?(Yes or [No]):
    Use same output port mapping for all input Ports?(Yes or [No]):
    Output port mapping:
       Input Port Number [1]?
       Bridge to all ports?(Yes or [No]):
       Bridge to  port 1  Yes or [No]:
       Bridge to  port 2  Yes or [No]:
       Bridge to  port 3  Yes or [No]:
       Bridge to  port 4  Yes or [No]:
       Bridge to  port 5  Yes or [No]:
       continue to another input port? (Yes or [No]): y
```

```
    Input Port Number [2]? 3
    Bridge to all ports?(Yes or [No]): y
    continue to another input port? (Yes or [No]): y
    Input Port Number [4]?
    Bridge to all ports?(Yes or [No]):
    Bridge to  port 1  Yes or [No]:
    Bridge to  port 2  Yes or [No]:
    Bridge to  port 3  Yes or [No]:
    Bridge to  port 4  Yes or [No]:
    Bridge to  port 5  Yes or [No]:
    continue to another input port? (Yes or [No]): n
  Source Address Filtering Applies? (Yes or No): y
  ASRT config>
```

Note: For any “Yes or No” question in the prompts, “No” is the default value. Press **Return** to accept the default value.

Exclude destination address ...

This prompt lets you set destination address filtering for that entry. Answering *yes* to the prompt causes filtering of any frames that contain this address as a destination address no matter which port it came from.

Use same output mapping...

Answering *yes* to this prompt lets you create one outgoing port map for all incoming ports rather than allowing for mapping to only specific ports. Answering *no* to this prompt causes further prompting (`Input Port Number [1]?`) to select each input port. From that specific input port prompt you can then create a unique port map for that input port.

Input Port 1, Port 2

Answering “No” to the previous prompt causes input port-by-input port prompting (`Input Port Number [1]?`) to select each input port and its associated outgoing bridge ports.

Bridge to all ports?

Answering *yes* to this prompt creates an outgoing port map that includes all ports. Thus, when a frame with this address as the destination address is received, it is forwarded to all outgoing forwarding ports except for the incoming port. The following are examples of how this is done according to the port map:

If a frame is received on *port 1* and the port map indicates 1 (for port 1), the frame is filtered.

If the same frame is received on *port 2* and the port map indicates 1 (for port 1), the frame is forwarded to port 1. If a frame is received on port 1 and the matching address entry’s port map indicates 1, 2, or 3, the frame is forwarded to ports 2 and 3.

If the port map indicates no port (NONE/DAF), the frame is filtered. This is known as destination address filtering (DAF).

If no address entry is found to match the received frame, it is forwarded to all the forwarding ports except for the source port.

Bridge to Port 1, Port 2, etc.

This prompt lets you associate an address entry with that specific bridge port. Answering *yes* maps the address to the specified port so that the port is included in that address entry’s port map. Answering *no* skips address mapping for that port.

continue to another bridge port?

This prompt lets you select the next input port to be configured.

Source address filtering

This allows for port-specific source address filtering (SAF). When SAF is applied (answer *yes* at the prompt), frames received with source addresses that match address entries in the filtering database that have source address filtering enabled will be discarded. This mechanism allows a network manager to isolate an end station by prohibiting its traffic to be bridged.

Enabling Destination Address Filtering For Entry

This example shows how to answer the command prompts to select destination address filtering for an entry:

```
ASRT config>add address 000000334455
Exclude destination address from all ports?(Yes or [No]): y
Source Address Filtering Applies? (Yes or [No]): y
ASRT config>
```

After adding the address entry, you can verify its status by using the **list range** command. The following example shows that no port map exists for that entry (in bold) and that destination address filtering (DAF) has been turned on.

```
ASRT config>list range
Start-Index [1]?
Stop-index [3]?
ADDRESS                 ENTRY TYPE      PORT MAP
=======                 ==========      ========
01-80-C2-00-00-00       REGISTERED      Input Port:  ALL PORTS
                                        Output ports:

00-00-00-22-33-44       PERMANENT       Input Port:  3
                                        Output ports: 1, 2
                                        Input Port:  4
                                        Output ports:  1, 2

00 00 00 33 44 55       PERMANENT       NONE/DAF
```

Output Port Map Created For Address Entry Having More Than One Input Port

This example shows how to answer the command prompts to create separate output port maps for an address entry that will have more than one input port.

```
ASRT config> add address 000000123456
Exclude destination address from all ports?(Yes or [No]):
Use same output port mapping for all input Ports?(Yes or [No]):
Input Port Number [1]? 1
Bridge to all ports ?(Yes or [No]):
Bridge to port 1 - Yes or [No]: y
Bridge to port 2 - Yes or [No]: y
Bridge to port 3 - Yes or [No]:
continue to another input port ? (Yes or [No]): y
Input Port Number [2]?
Bridge to all Ports?(Yes or [No]):
Bridge to Port 1 - Yes or [No]:
Bridge to port 2 - Yes or [No]:
Bridge to port 3 - Yes or [No]: y
continue to another input port ? (Yes or [No]):
Source Address Filtering Applies? (Yes or [No]):
ASRT config>
```

After adding the address entry, you can verify its status by using the **list range** command. The following example shows an entry (in bold) that has ports 1 and 2 as input ports and has separate port maps for both input ports. Source address filtering (SAF) has also been enabled.

```
ASRT config> list range
Start-Index [1]?
Stop-index [3]?
ADDRESS                 ENTRY TYPE      PORT MAP
=======                 ==========      ========
```

```
01-80-C2-00-00-00       REGISTERED      Input Port:  ALL PORTS
                                        Output ports:

01-80-C2-00-00-01       RESERVED        NONE/DAF

00-00-00-12-34-56       PERM/SAF        Input Port:  1
                                             Output ports: 1, 2
                                             Input Port:  2
                                             Output ports: 3
```

Single Output Port Map Created All Incoming Ports Associated With Address Entry

This example shows how to answer the command prompts to create a single output port map for all incoming ports associated with an address entry.

```
ASRT config> add address 000000556677
Exclude destination address from all ports?(Yes or [No]):
Use same output port mapping for all input Ports?(Yes or [No]): y
   Bridge to all ports?(Yes or [No]): n
   Bridge to  port 1 - Yes or [No]: y
   Bridge to  port 2 - Yes or [No]: y
   Bridge to  port 3 - Yes or [No]:
Source Address Filtering Applies? (Yes or [No]): y
ASRT config>
```

After adding the address entry, you can verify its status by using the **list range** command. The example below shows an entry (in bold) that has a single port map for all incoming ports. Source address filtering (SAF) has also been enabled.

```
ASRT config> list range
Start-Index [1]?
Stop-index [3]?
ADDRESS                 ENTRY TYPE      PORT MAP
=======                 ==========      ========
01-80-C2-00-00-00       REGISTERED      Input Port:  ALL PORTS
                                        Output ports:

01-80-C2-00-00-01       RESERVED        NONE/DAF

00-00-00-55-66-77       PERM/SAF        Input Port:  ALL PORTS
                                        Output ports:  1, 2
```

port *interface# port#*

Adds a LAN port to the bridging configuration. This command associates a port number with the interface number and enables that port's participation in transparent bridging.

Port Number Valid Values: 1 to 254

Port Number Default Value: none

Example: add a port

```
ASRT config> add port
Interface Number [0]?
Port Number [5]?
```

Delete

Use the **delete** command to delete the following information from your bridging configuration:

- Station address entries to the permanent database

Syntax:

delete address

port . . .

address *addr-value*

Deletes an address entry from the permanent database. The address is the MAC address of the desired entry. Enter the addr-value (in 12-digit hexadecimal format) of the entry to be deleted and press **Return**. Reserved multicast addresses cannot be deleted. If you attempt to delete an address entry that does not exist, you will receive the message

```
Record matching that address not found
```

Valid Values: X'0000 0000 0000' to X'FFFF FFFF FFFF'

Default Value: none

Example: **`delete address`**

port *port#*

Removes a port from a bridging configuration. Because the **enable bridge** command by default configures all LAN devices to participate in bridging, this command allows you to customize which devices should or should not participate in the bridging. The port number value normally is one greater than the interface number.

Example: **`delete port 2`**

Disable

Use the **disable** command to disable the following bridge functions:

- Bridging
- Propagation of Spanning Tree Explorer Frames
- Transparent (spanning tree) bridging function on a given port

Syntax:

disable bridge
stp
transparent . . .
tree

bridge

Disables bridging function entirely. This command does not remove previously configured bridging values, however.

Example: **`disable bridge`**

stp Disables the Spanning Tree Protocol on the bridge. The default is enabled.

Example: **`disable stp`**

transparent *port#*

Disables transparent bridging function on the given port.

Example: **`disable transparent 2`**

tree *port#*

Disables STP participation for the bridge on a per-port basis.

Example: **`disable tree 1`**

Note: Disabling STP on a per-port basis can produce network loops because of the existence of parallel bridges.

Enable

Use the **enable** command to enable the following bridging functions:
- Bridging
- Propagation of Spanning Tree Explorer Frames
- Transparent (Spanning Tree) bridging function on a given port

Syntax:

enable bridge . . .
stp
transparent . . .
tree

bridge
Enables transparent bridging function on all the LAN devices (interfaces) configured in the bridging device. The port numbers are assigned to each interface as the previous interface number plus 1. For example, if interface 0 is a LAN device its port number will be 1.

Example: `enable bridge`

stp Enables the spanning tree protocol on the bridge. This is the default.

Example: `enable stp`

transparent *port#*
Enables transparent bridging function on the given port. Under normal circumstances, this command is not necessary.

Example: `enable transparent`

```
Port Number [1]?
```

tree *port#*
Enables STP participation for the bridge on a per-port basis.

Example: `enable tree 1`

List

Use the **list** command to display information about the complete bridge configuration or to display information about selected configuration parameters.

Syntax:

list address
bridge
filtering . . .
permanent . . .
port . . .
protocol
range . . .

ASRT Configuration Commands (Talk 6)

address *addr value*

Reads an address entry from the permanent database. The addr value is the MAC address of the required entry. It can be an individual address, multicast address, or broadcast address. Permanent databases are not destroyed by the power off/on process and are immune to the aging settings. Permanent entries cannot be replaced by dynamic entries.

Valid Values: X'0000 0000 0000' to X'FFFF FFFF FFFF'

Default Value: none

Example: `list address 000000123456`

```
0000-00-12-34-56    PERMANENT  Input Port:  1
                               Output ports:  1, 2
                               Input port:  2
                               Output ports:  3
ASRT config>
```

Address

Address entry in 12-digit hexadecimal format.

Entry Type

Permanent

Indicates that the entry is permanent in nature and will survive power on/offs or system resets.

Reserved

Indicates that the entry is reserved by the IEEE 802.1d committee for future use. Frames destined to reserved addresses are discarded.

Registered

Indicates that the entry is meant for the bridge itself.

SAF Appears after the entry type if source address filtering has been configured.

Input Port

Displays the numbers of the input port or ports associated with that address entry.

Output Port

Displays the numbers of the output port or ports associated with that address entry. Displays "NONE/DAF" to indicate that destination address filtering applies because no ports have been selected to be associated with that address entry.

bridge

Lists all general information regarding the bridge.

filtering *datagroup-option*

The following general data groups can be displayed under the **list filtering** command:

All Displays all filtering database entries.

Ethertype

Displays Ethernet protocol type filter database entries.

SAP Displays SAP protocol filter database entries.

SNAP Displays SNAP protocol identifier filter database entries.

The following examples illustrate each of the **list filtering** display options.

Example 1: `list filtering all`

```
Ethernet type 0800 is routed on ports 1
IEEE 802.2 destination SAP 42 is routed on ports 1
IEEE 802 SNAP PID 00-00-00-08-00 is routed on ports 2-3
```

Descriptors used in explaining how packets are communicated include:

Routed
Describes packets passed to routing forwarder to be forwarded.

Filtered
Describes packets that are administratively filtered setting protocol filters that you set.

Bridged and routed
This describes a protocol identifier for which there is a protocol entity within the system that is not a forwarder. For example a link level echo protocol. Unicast packets from this protocol are bridged or locally processed if being sent to a registered address. Multicast packets are forwarded and locally processed for a registered multicast address.

All of these descriptors also apply to ARP packets with this Ethertype.

Example 2:

`list filtering ethertype`

```
Ethernet type (in hexadecimal), 0 for all [0]? 0800
Ethernet type 0800 is routed on ports 1
```

Example 3:

`list filtering sap`

```
SAP (in hexadecimal), 100 for all [100]? 42
IEEE 802.2 destination SAP 42 is routed on ports 1
```

Example 4:

`list filtering snap`

```
SNAP Protocol ID, return for all [00-00-00-00-00]?
IEEE 802 SNAP PID 00-00-00-08-00 is routed on ports 2-3
```

mapping *add-type type-field*
Lists specific address mapping for a given protocol.

Example: `list mapping SNAP`

```
PROTOCOL TYPE          GROUP ADDRESS          FUNCTIONAL ADDRESS
=============          =============          ==================
123456-7890            12-34-56-78-90-12      12:34:56:78:90:12
```

add-type
Choice of either DSAP, Ether (Ethernet), or SNAP.

type-field
Protocol type field:

- Destination Service Access Point (DSAP) protocol type is entered in the range 1–FE (hexadecimal).
- Ethernet (Ether) protocol type is entered in the range of 5DD–FFFF (hexadecimal).
- Subnetwork Access Protocol (SNAP) protocol type is entered in 10-digit hexadecimal format.

permanent
Displays the number of entries in the bridge's permanent database.

Example: `list permanent`

```
                    Number of Entries in Permanent Database:  17
```

port *port#*

Displays port information related to ports that are already configured. Port# selects the port you want to list. Specifying no number selects all ports.

Example: `list port`

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128: 2, (hex): 80-02
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only
Path Cost        : 0
```

Port ID

The ID consists of two parts: the port priority and the port number. In the example, 128 is the priority, and 1, 2, and 3 are the port numbers. In hexadecimal format, the low-order byte denotes the port number and the high-order byte denotes the priority.

Port state

Displays current state of the specified port or ports. This can be either ENABLED or DISABLED.

Port supports

Displays bridging method supported by that port (for example, transparent bridging).

Path Cost

Cost associated with the port which is used for possible root path cost. The range is 1 to 65535.

protocol

Displays bridge information related to the spanning tree protocol.

Note: Each of these bridge-related parameters is also described in detail in the previous chapter.

Bridge Identifier

8-byte value in ASCII format. If you did not set the bridge address prior to displaying this information, the low order 6 bytes will be displayed as zero, denoting that the default MAC address of a port is being used. When a bridge has been selected as the root bridge, the bridge max age and bridge hello time are transmitted by it to all the bridges in the network via the HELLO BPDUs.

Bridge-Max-Age

Maximum age (period of time) that should be used to time out spanning tree protocol-related information.

Bridge-Hello-Timer

Time interval between HELLO BPDUs.

Bridge-Forward-Delay

Time interval used before changing to another state (should this bridge become the root).

range *start-index stop-index*

Reads a range of address entries from the permanent database. To specify this, first determine the size of the database by using the **list permanent** command. From this value you can then determine a “start index” value for your entry range. The start index is in the range from 1 to the size of the database. You can then choose a “stop index” for displaying a limited

number of entries. This input is optional. If you do not specify the stop index, the default value is the size of the database.

Address entries contain the following information:

Example: `list range`

```
Start-Index [1]? 1
Stop-index [17]? 6
ADDRESS                 ENTRY TYPE      PORT MAP

=======                 ==========      ========
01-80-C2-00-00-00       REGISTERED      Input Port:  ALL PORTS
                                        Output ports:

01-80-C2-00-00-01       RESERVED        NONE/DAF
01-80-C2-00-00-02       RESERVED        NONE/DAF
01-80-C2-00-00-03       RESERVED        NONE/DAF
01-80-C2-00-00-04       RESERVED        NONE/DAF
01-80-C2-00-00-05       RESERVED        NONE/DAF
```

Address

6-byte MAC address of the entry.

Type of Entry

Specifies one of the following types:

- Reserved - entries reserved by the IEEE 802.1d committee
- Registered - entries consist of unicast addresses belonging to proprietary communications hardware attached to the box or multicast addresses enabled by protocol forwarders
- Permanent - entries entered by the user in the configuration process which survive power on/offs or system resets
- Static - entries entered by the user in the monitoring process that do not survive power on/offs or system resets and are ageless
- Dynamic - entries “learned” by the bridge “dynamically” that do not survive power on/offs or system resets and that have an “age” associated with the entry
- Free - locations in database that are free to be filled by address entries

Port Map

Displays outgoing port map for all incoming ports.

Set

Use the **set** command to set certain values, functions, and parameters associated with the bridge configuration. These include:

- Aging time for dynamic address entries in the filtering database
- Bridge address
- MAC service data unit (MSDU) size
- Spanning tree protocol bridge and port parameters
- Size of the bridge filtering database

Syntax:

set	age
	bridge
	filtering
	port
	protocol bridge

protocol port . . .

age *seconds resolution*

Sets the time for aging out dynamic entries in the filtering database when the port with the entry is in the forwarding state. This age is also used for aging RIF entries in the adaptive database in the case of an SR-TB bridge personality.

Enter the required value after each prompt and press **Return**.

Aging Time Valid Values: 10 to 1000000

Aging Time Default Value: 30

The resolution value specifies how often dynamic entries in the filtering database should be scanned to determine if they have exceeded their age limit as set by the aging timer.

Resolution Valid Values: 1 to 60 seconds

Resolution Default Value: 5 seconds

Example: `set age`

```
seconds [300]  ?  400
resolution [5] ?  6
```

bridge *bridge-address*

Sets the bridge address. This is the low-order 6-octet bridge address found in the bridge identifier. By default, the bridge-addr-value is set to the medium access control (MAC) address of the lowest-numbered port at initialization time. You can use this command to override default address and enter your own unique address.

Enter *tb* to specify that the transparent bridge (tb) bridge address is to be affected.

Note: Each bridge in the network must have a unique address for the spanning tree protocol to operate correctly.

Attention: In cases where a serial line interface is the lowest numbered port, it is mandatory to use this command so that the bridge will have a unique address when restarted. This process is necessary because serial lines do not have their own MAC address.

At the prompt, enter the bridge address in 12-digit hexadecimal format and press **Return**.

If you enter the address in the wrong format you will receive the message `Illegal Address`. If you enter no address at the prompt you will receive the message `Zero length address supplied` and the bridge will maintain its previous value. To return the bridge address to the default value, enter an address of all zeros.

Valid Values: 12 hexadecimal digits

Do not use dashes or colons to separate each octet. Each bridge in the network must have a unique address for the spanning tree protocol to operate correctly.

Default Value: 000000000000

Example: `set bridge`

```
Bridge Address (in 12-digit hex)[]?
```

filtering *database-size*

Sets the number of entries that can be held in the bridge filtering database.

Default Value: 1024 times the number of bridge ports.

For more information, see the **list filtering** command on page 266.

Example: `set filtering`

```
database-size [2048]?
```

port *block* **or** *disable*

Begins the port's participation in the spanning tree protocol. This is done by entering a status value of "block." This places the port in the "blocked" status as a starting point. The actual state of the port will later be determined by the spanning tree protocol as it determines its topology. Entering a status value of "disable" removes the port from participating in the spanning tree.

Example: `set port block`

```
Port Number [1]?
```

protocol *bridge* **or** *port*

Modifies the spanning tree protocol bridge or port parameters for a new configuration, or tunes the configuration parameters to suit a specific topology.

Enter "bridge" as the option to modify bridge parameters. The bridge-related parameters that can be modified with this command are described below.

When setting these values, make sure that the following relationships exist between the parameters or the input will be rejected:

2 X (Bridge Forward Delay - 1 second) ≥ Bridge Maximum Age
Bridge Maximum Age ≥ 2 X (Bridge Hello Time + 1 second)

Example: `set protocol bridge tb`

```
Bridge Max-Age [20]  25
Bridge Hello Time [2]   3
Bridge Forward Delay [15] 20
Bridge Priority [32768]    1
```

Bridge Maximum Age

Maximum age (period of time) that should be used to time out spanning tree protocol-related information.

When this bridging device is selected as the root bridge in a spanning tree, the value of this parameter specifies how long other active bridges are to store the configuration bridge protocol data units (BPDUs) they receive. When a BPDU reaches its maximum age limit without being replaced, the active bridges in the network discard it and assume that the root bridge has failed. A new root bridge is then selected.

Dependencies

The setting of this parameter may be affected by the setting of the Bridge Hello Time parameter. In addition, the setting of this parameter may affect the setting of the Bridge Forward Delay parameter.

Valid Values: 6 to 40 seconds

Default Value: 20 seconds

Bridge Hello Timer

Time interval between HELLO BPDUs.

When this bridging device is selected as the root bridge in a spanning tree, this parameter specifies how often this bridge transmits configuration bridge protocol data units (BPDUs). BPDUs contain information about the topology of the spanning tree and reflect changes to the topology.

Dependencies

The setting of this parameter may affect the setting of the Max age parameter.

Valid Values: 1 to 10 seconds

Default Value: 2

Bridge Forward Delay

Time interval used before changing to another state (should this bridge become the root).

When this bridging device is selected as the root bridge in a spanning tree, the value of this parameter specifies how long active ports in all bridges remain in a *listening state*. When the forward delay time expires, ports in the listening state go into the *forwarding state*. State changes occur as a result of changes in the topology of the spanning tree, such as when an active bridge fails or is shut down.

The root bridge conveys this value to all bridges. This process ensures that all bridges are consistent between changes.

Valid Values: 4 to 30 seconds

Default Value: 15

Bridge Priority

A high-order 2-octet bridge address found in the Bridge Identifier - either the MAC address obtained from the lowest-numbered port or the address set by the **Set Bridge** command.

The bridge priority indicates the chances that this bridge will become the root bridge of the spanning tree. The lower the numerical value of the bridge priority parameter, the higher the priority of the bridge and the more likely it is to be chosen. The spanning tree algorithm chooses the bridge with the lowest numerical value of this parameter to be the root bridge.

Valid Values: 0 to 65535

Default Value: 32768

Enter **port** as the option to modify the spanning tree protocol port parameters. Enter the desired value at each prompt and press **Return**.

Example: `set protocol port`

```
Port Number [1] ?
Port Path-Cost (0 for default) [0] ?   1
Port Priority [128] ?   1
```

Port Number
Bridge port number; selects the port for which the path cost and port priority will be changed.

Path Cost
Cost associated with the port, which is used for possible root path cost.

Each port interface has an associated path cost, which is the relative value of using the port to reach the root bridge in a bridged network. The spanning tree algorithm uses the path cost to compute a path that minimizes the cost from the root bridge to all other bridges in the network topology.

This parameter specifies the cost associated with passing frames through this port interface, should this bridging device become the root bridge. Factor this value in when determining spanning tree routes between any two stations. A value of 0 instructs the bridging device to automatically calculate a path cost for this port using its own formula.

Valid Values: 1 to 65535

Default Value: 0 (means the cost will be calculated automatically)

Port Priority
Identifies port priority for the specified port. This is used by the spanning tree algorithm in making comparisons for port selection (which port offers the lowest cost path to the root bridge) and blocking decisions.

Valid Values: 0 to 255

Default Value: 128

VLANS

Use the **vlans** command to access the VLAN configuration prompt. VLAN configuration commands are entered at this prompt. See “Dynamic Protocol Filtering (VLANS) Configuration Commands” for an explanation of each of these commands.

Syntax:

vlans

Dynamic Protocol Filtering (VLANS) Configuration Commands

This section explains all of the VLAN configuration commands. These commands let you configure protocol and IP multicast VLANs.

See “Dynamic Protocol Filtering VLANs” on page 255 for additional information about VLANs.

Configuration commands for the ASRT bridge are entered at the `ASRT VLAN config>` prompt. This prompt is accessed by entering the **vlans** command at the `ASRT config>` prompt. The following table shows the VLAN filtering configuration commands.

Table 53. VLAN Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Adds the definition of a new VLAN filter
Change	Changes VLAN filtering parameters for an indicated VLAN
Delete	Deletes the selected VLAN filters
Disable	Disables VLAN filtering on the selected VLANs
Enable	Enables VLAN filtering on the selected VLANs
List	Displays all information associated with the selected VLAN filters
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **Add** command to define a new VLAN filter. See “Required Static Configurations” on page 256 for additional information.

Syntax:

add ip

ip-multicast

ipx

netbios

sliding-window

Example 1: add ip

```
IP Address [0.0.0.0]? 9.2.3.4
Subnet Mask [255.0.0.0]?
Configure this VLAN on Specific Ports? [No]:
Age (expiration in minutes,0=infinity) [10000]? 0
Enable IP-Cut-Through from this VLAN? [Yes]:
Enable IP-Cut-Through to this VLAN? [Yes]:
Track Active MAC Addresses on this VLAN? [No]:
Enable This Filter? [Yes]:
VLAN Name (32 chars max) []? IP 9.x.x.x
VLAN 'IP 9.x.x.x' (IP subnet 9.0.0.0) successfully added
```

If some ports should not be configured as Auto-Detect and Include, then the port can be manually configured.

Example 2: add ip-multicast

```
IP Multicast Address [0.0.0.0]? 230.1.1.1
Configure Specific Ports? [No]:
Age (expiration in minutes,0=infinity) [10]? 0
Track Active MAC Addresses on this VLAN? [No]:
Enable This Filter? [Yes]:
VLAN Name (32 chars max) []? IPmcast01
VLAN 'IPmcast01' (IP Multicast 230.1.1.1) successfully added
```

Example 3: add ipx

```
Network Number (in 8-digit hex) (1 - FFFFFFFE) [1]? 2FF
Configure this VLAN on Specific Ports? [No] y
Configure VLAN on port 1 (Include, Exclude, or Auto-Detect) [A]?
Configure VLAN on port 2 (Include, Exclude, or Auto-Detect) [A]? e
Age (expiration in minutes,0=infinity) [5000]?
Track Active MAC Addresses on this VLAN? [No]:
```

```
Enable This Filter? [Yes]:
VLAN Name (32 chars max) []? IPX 2FF
VLAN 'IPX 2FF' (IPX network 0x2FF) successfully added
```

A description of each parameter follows:

IP Address
This prompt allows you to enter the IP address of the IP subnet whose traffic will be dynamically filtered to create this VLAN. This value, after the subnet mask is applied, is what will be saved and referenced in other VLAN commands.

Subnet Mask
This is the subnet mask that will be applied to the input IP Address to create the IP subnet value used to detect traffic for this VLAN.

IP Multicast address
This is the IP group address whose multicast traffic will be filtered to create this VLAN.

Note: A VLAN for 224.0.0.1 (the all IP hosts address) is created during initialization and is used to configure IP multicast VLANs that are auto-created when an IGMP report frame is detected and the 224.0.0.1 VLAN is enabled. See “Auto-created IP Multicast VLANs” on page 257 for additional information about auto-created IP multicast VLANs.

Valid Values: 224.0.1.0 - 239.255.255.255

Default Value: none

Network Number
This prompt allows you to enter the IPX network ID number whose traffic will be dynamically filtered to create this VLAN.

Sliding Window Filter Base
Determines whether the base for the offset is the first byte of the destination MAC address or the first byte of the frame’s information field.

Valid Values: mac or info

Default Value: mac

Sliding Window Filter Offset
Sets the byte offset into the frame where the comparison with the mask and value begins.

Valid Values: 0 - 255

Default Value: 0

Sliding Window Filter Value
The value used for comparing the sliding window filter.

A frame “matches” a sliding window filter if the octet pattern (whose start is determined by the *Sliding Window Filter Base* and *Sliding Window Filter Offset*) ANDED with the *Sliding Window Filter Mask* equals this *Sliding Window Filter Value* ANDED with the *Sliding Window Filter Mask*.

Valid Values: Any octet string of length 1 - 32

Default Value: None

Sliding Window Filter Mask
The mask used for comparing the sliding window filter.

Valid Values: Any octet string of length 1 - 32

Default Value: None

Configure

Answering “No” to this prompt causes all bridge ports to be set to the default value of Auto-Detect and Include. Answering *yes* to this prompt causes further prompting to select the desired port inclusion mode for each bridge port.

The modes are:

- Auto-Detect and Include (the default mode that requires that traffic from this vlan be received on the port before being included in the VLAN forwarding domain).
- Include Always (to always include this port in the forwarding domain regardless of received traffic)
- Exclude Always (to always exclude this port from the forwarding domain regardless of received traffic).

Age The amount of time, in minutes, that an Auto-Detect port will remain in the forwarding state in the absence of traffic received from that port for this VLAN. Entering a value of zero means that ports auto-detected will never expire and be removed from the forwarding domain.

If MAC address tracking is enabled for a VLAN, the aging time also determines when a MAC address is no longer considered a member of the VLAN in the absence of traffic received from that MAC address.

Valid Values: 0 to 4 294 967 295

Default Value

IP subnet
10 000 minutes

IP multicast
10 minutes

IPX Network
10 minutes

NetBIOS
5 000 minutes

Sliding Window
5000 minutes

Enable IP-Cut-Through Transmission Status
Answering *yes* will allow forwarding of IP traffic from devices on this VLAN to devices on other VLANs that have IP-Cut-Through reception enabled. See “IP-Cut_Through Considerations” on page 256 for additional information.

Enable IP-Cut-Through Reception Status
Answering *yes* will allow IP traffic to be forwarded to devices on this VLAN from devices on other VLANs that have IP-Cut-Through transmission enabled. See “IP-Cut_Through Considerations” on page 256 for additional information.

Track Active MAC Addresses

Answering *yes* causes source MAC addresses from transmissions on this VLAN to be saved. These learned addresses can be displayed with the **show-members** command. Learned addresses will be aged out with the aging timer for this VLAN.

VLAN Filter Status

Answering *yes* will enable dynamic filtering for this VLAN. Answering “No” means that no filtering will be done on traffic from members of this VLAN.

VLAN Name

This prompt lets you define a name for this VLAN that can be used with all VLAN commands. A VLAN name is required for MAC address, port-based, and sliding window VLANs.

This name must be unique among all VLANs of all types within the ASRT bridge. This name consists of up to 32 characters and can include spaces.

Change

Use the change command to change the configuration parameters associated with a particular VLAN. The VLAN to change can be chosen by explicitly specifying the subnet or by selecting the VLAN from a list with the *by-name* option. This command invokes the same prompts used with the add command. The current parameter values will be displayed as the default and can be maintained by simply pressing **Return**.

Syntax:

change by-name

ip subnet address

ip-multicast

ipx network number

netbios

sliding-window

Example: change ip

```
IP Address [9.0.0.0]?
Configure Specific Ports? [No]:
Age (expiration in minutes,0=infinity) [0]? 300
Enable IP-Cut-Through from this VLAN? [Yes]:
Enable IP-Cut-Through to this VLAN? [Yes]:
Track Active MAC Addresses on this VLAN? [No]:
Enable This Filter? [Yes]:
VLAN Name (32 chars max) [IP 9.x.x.x]?
VLAN 'IP 9.x.x.x' (IP subnet 9.0.0.0) successfully changed
```

Delete

Use the **delete** command to delete a particular VLAN filter, all VLAN filters of a particular type, or all defined VLAN filters. If you are deleting a single filter, you can choose the VLAN to be deleted by selecting the VLAN from a list using the *by-name* option.

Syntax:

delete by-name

ip all

ip subnet *subnet address*

ip-multicast all

ip-multicast by-name

ipx all

ipx network *network-number*

netbios

sliding-window all

sliding-window by-name

all

Example 1: del ip subnet 9.0.0.0

```
VLAN 'IP 9.x.x.x' (IP subnet 9.0.0.0) deleted
```

Example 2: del ipx all

```
Are you sure you want to delete ALL IPX VLANS? [No]: y
All IPX VLANS deleted
```

Disable

Use the **disable** command to disable a particular VLAN filter, all VLAN filters of a particular type, or all defined VLAN filters. If disabling a single filter, the VLAN to be disabled can be chosen by selecting the VLAN from a list using the *by-name* option.

Syntax:

disable

by-name

ip all

ip subnet *subnet-address*

ip-multicast all

ip-multicast by-name

ipx all

ipx network *network-number*

netbios

sliding-window all

sliding-window by-name

all

Example: disable ip subnet 220.5.3.0

```
VLAN 'Building #4' (IP subnet 220.5.3.0) now disabled
```

Enable

Use the **enable** command to enable a particular VLAN filter, all VLAN filters of a particular type, or all defined VLAN filters. If you are enabling a single filter, you can choose the VLAN to be enabled by selecting the VLAN from a list using the *by-name* option.

Syntax:

enable	by-name
	ip all
	ip subnet *subnet-address*
	ip-multicast all
	ip-multicast by-name
	ipx all
	ipx network *network-number*
	netbios
	sliding-window all
	sliding-window by-name
	all

Example: enable by-name

```
 Choice of VLAN:
       VLAN type    Identifier         VLAN Name
       =========    ==========         =========
    (1) IP           9.0.0.0            IP 9.x.x.x
    (2) IP           220.5.3.0          Building #4
    (3) IPX          0x2FF              Ethernet A
    (4) IPX          0x3FF              Ethernet B
Enter Selection [1]? 3
VLAN 'Ethernet A' (IPX Network 0x2FF) now enabled
```

List

Use the **list** command to list the configuration information about a particular VLAN filter, all VLAN filters of a particular type, or all defined VLAN filters. If you are listing a single filter, you can choose the VLAN to be listed can be chosen by selecting the VLAN from a list using the *by-name* option.

Syntax:

list	by-name
	ip all
	ip subnet *subnet-address*
	ip-multicast all
	ip-multicast by-name
	ipx all
	ipx network *network-number*
	netbios
	sliding-window all

sliding-window by-name

all

Example 1: list ip subnet 9.0.0.0

```
Subnet Address              = 9.0.0.0
Subnet Mask                 = 255.0.0.0
Bridge Port 1 (Interface 0) = Auto-Detect and Include
Bridge Port 2 (Interface 1) = Always Exclude
Age (expiration in minutes) = 300
IP-Cut-Through Status:
   Transmit From This VLAN  = Enabled
   Reception By This VLAN   = Enabled
Tracking of MAC Addresses   = Disabled
VLAN Filter State           = Enabled
VLAN Name                   = IP 9.x.x.x
```

Example 2: list ipx all

```
----------- IPX VLANS -------------------------------
IPX Network Number          = 0x2FF
Bridge Port 1 (Interface 0) = Auto-Detect and Include
Bridge Port 2 (Interface 1) = Always Exclude
Age (expiration in minutes) = Never Expires
         Tracking of MAC Addresses   = Disabled
VLAN Filter State           = Enabled
VLAN Name                   = Ethernet A
+++++++++++++++++++++++++++++++++++++++++++++++++++++++
IPX Network Number          = 0x3FF
Bridge Port 1 (Interface 0) = Auto-Detect and Include
Bridge Port 2 (Interface 1) = Auto-Detect and Include
Age (expiration in minutes) = 5000
IP-Cut-Through Status:
   Transmit From This VLAN  = Enabled
   Reception By This VLAN   = Enabled
Tracking of MAC Addresses   = Disabled
VLAN Filter State           = Disabled
VLAN Name                   = Ethernet B
```

Accessing the ASRT Monitoring Environment

To access the ASRT monitoring environment, enter the **protocol asrt** command at the + (GWCON) prompt:

```
+protocol asrt
ASRT>
```

ASRT Monitoring Commands

This section describes the ASRT monitoring commands. These commands allow you to view and modify parameters from the active monitoring. Information you modify with the monitoring commands is reset to the SRAM configuration when you restart the bridging device.

You can use these commands to temporarily modify the configuration without losing configuration information in the bridge memory. The ASRT> prompt is displayed for all ASRT monitoring commands.

Monitoring and dynamic reconfiguration VLANS commands are entered at the VLAN> monitoring prompt. The VLAN> command is accessed by entering the **VLANS** command explained later in this chapter.

Note: For commands requiring you to enter MAC Addresses, the addresses can be entered in the following formats:

IEEE 802 canonical bit order
00-00-00-12-34-56
IEEE 802 canonical bit order (shorthand format)
000000123456

Table 54 shows the ASRT monitoring commands.

Table 54. ASRT Monitoring Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Adds permanent (static) address entries to the bridging device’s permanent database.
Cache	Displays cache entries for a specified port.
Delete	Deletes MAC addresses entries from the bridging device database.
Flip	Flips MAC address from canonical to 802.5 (noncanonical or IBM) bit order.
List	Displays information about the complete bridge configuration or about selected configuration options.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **add** command to add static address entries and destination address filters to the bridging device’s database. These additions to the database are lost when you restart the device.

Syntax:

add static-entry

static-entry *mac_address input_port [output_ports]*
Adds static address entries to the bridging device’s permanent database. Enter the command followed by the MAC address of the static entry and the input port number (an optional output port number may also be entered). To create a static entry with multiple port maps (1 per input port), use this command several times.

Example: `add static-entry`

```
MAC address [00-00-00-00-00-00]?  400000012345
Input port, 0 for all [0]? 2
Output port, 0 for none [0]? 3
Output port, 0 to end [0]?
```

Cache

Use the **cache** command to display the contents of a selected bridging-port routing cache. If the port does not possess a cache you will see the message `Port X does not have a cache`.

Syntax:

cache *port#*

Example: `cache`

```
Port number [1]? 3

MAC Address    MC*   Entry Type       Age  Port(s)

00-00-93-00-C0-D0   PERMANENT          0  3 (TKR/1)
00-00-00-11-22-33   STATIC             0  3 (TKR/1)
```

MAC Address
6-byte MAC address of the entry.

Entry Type
Specifies one of the following address entry types:

Reserved - entries reserved by the IEEE 802.1d Standard.

Registered - entries consist of unicast addresses belonging to proprietary communications hardware attached to the box or multicast addresses enabled by protocol forwarders.

Permanent - entries entered by the user in the configuration process which survive power on/offs or system resets.

Static - entries entered by the user in the monitoring process which do not survive power on/offs or system resets and are not effected by the aging timer.

Dynamic - entries “learned” by the bridge “dynamically” which do not survive power on/offs or system resets and which have an “age” associated with the entry.

Free - locations in database that are free to be filled by address entries.

Unknown - entry types unknown to the bridge. May be possible bugs and/or illegal addresses.

Age Age in seconds of each dynamic entry. Age is decremented at each resolution intervals.

port(s)
Specifies the port number associated with that entry and displays the interface name (this will always be that of the interface having the cache).

Delete

Use the **delete** command to delete station (including MAC) address entries from the device’s permanent database.

Syntax:

delete mac-address

Example: `delete 00-00-93-10-04-15`

Flip

Use the **flip** command to view specific MAC addresses in the canonical and noncanonical format by “flipping” the address bit order. This command is useful for translating IEEE 802.5 addresses in their typical noncanonical format to the canonical format universally used by the bridge monitoring and ELS (and vice versa).

Syntax:

flip *MAC-address*

Example: `flip`

```
MAC address [00-00-00-00-00-00]? 00-00-00-33-44-55
IEEE 802 canonical bit order:    00-00-00-33-44-55
IBM Token-Ring native bit order: 00:00:00:CC:22:AA
```

List

Use the **list** command to display information about the bridging device configuration or to display information about selected configuration or bridging options.

Syntax:

list bridge . . .
database . . .
filtering . . .
port
spanning-tree-protocol . . .
transparent . . .

bridge

Lists all general information regarding the bridge device configuration.

list bridge active
all

active Displays information about only the active bridge devices.

all Displays information about all bridge devices.

Example: `list bridge active`

Bridge ID
Unique ID used by the spanning tree algorithm in determining the spanning tree. Each bridge in the network is assigned a unique bridge identifier. The bridge priority is displayed in decimal followed by the hex address.

Bridge State
Indicates whether bridging is enabled or disabled.

Bridge Type
Displays the configured bridge type. This is displayed as NONE, TB, or ASRT.

Number of Ports
Displays the number of ports configured for that bridge.

Port Specifies a user defined number assigned to an interface by the **add port** command.

Interface
Identifies devices connected to a network segment through the bridge.

State Indicates the current state of the port. This is displayed as UP or DOWN.

MAC address
Displays the MAC address associated with that port in canonical bit order.

Modes
Displays the bridging mode for that port. T indicates transparent bridging. SR indicates source routing. A indicates adaptive bridging.

MSDU Specifies the maximum frame (data unit) size (including the MAC header but not the FCS field) the bridge can transmit and receive on this interface.

Segment
Displays the source routing bridge segment number assigned to that port (if any).

database *datagroup-option*
Lists the contents of transparent filtering databases. There are a number of datagroups which can be chosen to be displayed under the list database command. These include the following:

- All - Displays the entire transparent bridging database.
- Dynamic - Displays all dynamic (learned) address database entries.
- Local - Displays all local (reserved) address database entries.
- Permanent - Displays all permanent address database entries.
- Port - Displays address entries for a specific port.
- Range - Displays a range of database entries from the total transparent bridging filtering address database. A starting and ending MAC address is given to define the range. All entries falling within this range will be displayed.
- Sorted - Displays a sorted list of database entries.
- Static - Displays static entries from the address database.
- Unsorted - Displays an unsorted list of database entries.

The following examples break down the list database command options. The first example also shows the related output.

Example: `list database all`

Note: The following fields are displayed for all of the **list database** command options.

MAC Address
Specifies the address entry in 12-digit hex format (canonical bit order).

MC* An asterisk following an address entry indicates that the entry has been flagged as a multicast address.

Entry Type
Specifies one of the following types:

Reserved
Entries reserved by the IEEE 802.1d standard.

Registered
Entries consist of unicast addresses belonging to interfaces participating in the bridge or multicast addresses enabled by protocol forwarders

Permanent
Entries entered by the user in the configuration process which survive power on/offs or system resets

Static Entries entered by the user in the monitoring process which do not survive power on/offs or system resets and are ageless.

Dynamic
Entries “learned” by the bridge “dynamically” which do not survive power on/offs or system resets and which have an “age” associated with the entry

Free This type is not used and should not be normally be seen except in occasional “race” conditions between the monitoring and the bridge.

Unknown
Unknown entry type. May indicate a software bug. Report the hex entry type to Customer Service.

filtering *datagroup-option*

Displays general information about the bridge’s protocol filtering databases. There are a number of general datagroups which may be displayed under the **list filtering** command. These include the following:

- All - Displays all filtering database entries.
- Ethertype - Displays Ethernet protocol type filter database entries.
- SAP - Displays SAP protocol filter database entries.
- SNAP - Displays SNAP protocol identifier filter database entries.

The following examples break down each of the list filtering display options.

Example: `list filtering all`

```
Ethernet type 0800 is routed on ports 1
IEEE 802.2 destination SAP 42 is routed on ports 1
IEEE 802 SNAP PID 00-00-00-08-00 is routed on ports 2-3
```

Descriptors used in explaining how packets are communicated include the following:

- Routed - Describes packets which are passed to routing forwarder to be forwarded
- Filtered- Describes packets which are administratively filtered by the user setting protocol filters
- Bridged and routed - This describes a protocol identifier for which there is a protocol entity within the system which is not a forwarder. An example of this would be a link level echo protocol. Unicast packets from this protocol are bridged or locally processed if being sent to a registered address. Multicast packets are forwarded and locally processed for a registered multicast address.

All of the descriptors just explained also apply to ARP packets with this Ethertype.

Example: `list filtering ethertype`

```
Ethernet type (in hexadecimal), 0 for all [0]? 0800
Ethernet type 0800 is routed on ports 1
```

Example: `list filtering SAP`

```
SAP (in hexadecimal), 100 for all [100]? 42
IEEE 802.2 destination SAP 42 is routed on ports 1
```

Example: `list filtering SNAP`

```
SNAP Protocol ID, return for all [00-00-00-00-00]?
IEEE 802 SNAP PID 00-00-00-08-00 is routed on ports 2-3
```

port *port#*

Displays port information.

Example: list port

```
Port Id (dec)    : 128: 3, (hex): 80-03
Port State       : Forwarding
STP Participation: Enabled
Port Supports    : Transparent Bridging Only
Assoc Interface #/name : 5/Eth/1
```

Port Specifies a user defined number assigned to an interface by the **add port** command.

Interface
Identifies devices connected to a network segment through the bridge.

State Indicates the current state of the port. This is displayed as UP or DOWN.

MAC address
Displays the MAC address associated with that port in canonical bit order.

Modes
Displays the bridging mode for that port. **T** indicates transparent bridging. **SR** indicates source routing. **A** indicates adaptive bridging.

MSDU Specifies the maximum frame (data unit) size (including the MAC header but not the FCS field) the bridge can transmit and receive on this interface.

Segment
Displays the source routing bridge segment number assigned to that port (if any).

spanning-tree protocol *datagroup-option*

- Displays spanning tree protocol information. The spanning tree protocol is used by the transparent bridge to form a loop-free topology. There are a number of general datagroup options which may be displayed under the **list spanning-tree-protocol** command. These include the following:
  - Configuration - Displays information concerning the spanning tree protocol.
  - Counters - Displays the spanning tree protocol counters.
  - State - Displays the current spanning tree protocol state information.
  - Tree - Displays the current spanning tree information including port, interface, and cost information.

  The following examples illustrate each of the list spanning-tree-protocol display options.

Example: `list spanning-tree-protocol configuration`

```
Bridge ID (prio/add):  32768/0000-93-00-84-EA
Bridge state:          Enabled
Maximum age:           20 seconds
```

```
Hello time:            2 seconds
Forward delay:         15 seconds
Hold time:             1 seconds
Filtering age:         320 seconds
Filtering resolution:  5 seconds

Port  Interface  Priority  Cost   State
  4   Eth/1      128     100     Enabled
128   Tunnel     128     65535   Enabled
```

Example: `list spanning-tree-protocol counters`

```
             Time since topology change (seconds)         0
             Topology changes:                            1
             BPDUs received:                              0
             BPDUs sent:                              14170

Port  Interface  BPDUs received  BDPU input overflow  Forward transitions

   1  TKR/1             0              0               1
```

Example: `list spanning-tree-protocol state`

```
Designated root (prio/add):    32768/00-00-93-00-84-EA
Root cost:                     0
Root port:                     Self
Current (root) maximum age:    20 seconds
Current (root) hello time:     2 seconds
Current (root) Forward delay:  15 seconds
Topology change detected:      FALSE
Topology change:               FALSE

Port  Interface   State
  4  Eth/1       Forwarding
```

Example: `list spanning-tree-protocol tree`

```
Port                  Designated   Desig.         Designated   Des.
No.  Interface             Root     Cost             Bridge   Port

 2  ATM/0:0:48  0/00-00-00-00-00-00    0     0/00-00-23-45-00-00 80-00
```

transparent

Displays the status of the transparent bridging function.

Dynamic Protocol Filtering (VLANS)

The VLAN monitoring commands are a superset of the VLAN configuration commands. However, instead of updating the SRAM configuration records immediately, they change the behavior of VLANs in real-time. Changes made through the monitoring can be optionally saved to SRAM. Also, the configuration in SRAM can be loaded and used without requiring a reboot.

Monitoring commands for the ASRT bridge are entered at the `ASRT VLAN>` prompt. This prompt is accessed by entering the **vlans** command at the `ASRT>` prompt. The following table shows the VLAN monitoring commands.

Table 55. VLAN Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Adds the definition of a new VLAN filter
Change	Changes VLAN filtering parameters for an indicated VLAN
Delete	Deletes the selected VLAN filters
Disable	Disables VLAN filtering on the selected VLANs
Enable	Enables VLAN filtering on the selected VLANs
List	Displays all information associated with the selected VLAN filters
Load	Loads and uses the VLAN configuration currently in SRAM
Reset-Counters	Resets all counters associated with the selected VLAN filters

Table 55. VLAN Monitoring Command Summary (continued)

Command	Function
Save	Saves the current runtime configuration to SRAM
Show-members	Displays learned MAC addresses for a selected VLAN
Show-vlans	Lists the enabled VLANs of which a particular MAC address is a member
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

For a description of the **Add**, **Change**, **Delete**, **Disable**, and **Enable** commands, see “Dynamic Protocol Filtering (VLANS) Configuration Commands” on page 273.

List Use the list command to list the current real-time configuration for a particular VLAN filter, all VLAN filters of a particular type, or all defined VLAN filters. If listing a single filter, the VLAN to list can be chosen by selecting the VLAN from a list with the *by-name* option. The resulting output includes both configuration parameters and VLAN counters.

Syntax:

list by-name

ip all

ip subnet *subnet address*

ip-multicast all

ip-multicast by-name

ipx all

ipx network *network number*

netbios

sliding-window all

sliding-window by-name

all

Example:

```
vlans config>list ip subnet 9.0.0.0

   Subnet Address              = 9.0.0.0
   Subnet Mask                 = 255.0.0.0
   Port 1 (Interface 0) = Auto-Detect and Include, Forwarding
   Port 2 (Interface 1) = Always Exclude,          Not Forwarding
   Age (expiration in minutes) = 300
   IP-Cut-Through Status:
      Tx From This VLAN    = Enabled  Reception By This VLAN = Disabled
      Packets Transmitted  = 25       Packets Received       = 0
      Tx Packets Discarded = 0        Rx Packets Discarded   = 14
   Tracking of MAC Addresses  = Disabled
   VLAN Status                = Enabled
   Packets Processed          = 43
   Discards Due To Exclusion  = 13
   VLAN Name                   = IP 9.x.x.x
```

A description of the VLAN counters follows:

Packets Transmitted
Total number of IP packets successfully cut through from this VLAN.

Packets Received
Total number of IP packets successfully cut through to this VLAN.

Tx Packets Discarded
Number of IP packets that were intended to be cut through from this VLAN, but were discarded due to IP-Cut-Through transmission being disabled. Packets from ports configured as Always Exclude are not included in this count.

Rx Packets Discarded
Number of IP packets that were intended to be cut-through to this VLAN, but were discarded due to IP-Cut-Through reception being disabled.

Packets Processed
Total number of packets processed by this VLAN's forwarding logic. This includes all packets forwarded and discarded. For IP Multicast VLANs, this number includes IGMP Reports and matching IP Multicast frames. For the IP Multicast auto-creation VLAN (group 224.0.0.1), this counter indicates the number of received IGMP Query packets from multicast devices.

Discards Due To Exclusion
Number of packets received matching this VLAN on ports configured as Always Exclude for this VLAN.

Load Use the load command to load and immediately use the VLAN configuration stored in SRAM. This will overwrite any configuration changes that may have been made via monitoring since the last save. All timers and counters associated with VLANs will be reset.

Syntax: load

Example: load

```
Warning: This process will overwrite your current configuration.
Are you sure you want to load the VLAN configuration from SRAM? [No] y
VLAN configuration loaded
```

Reset-Counters
Use the reset-counters command to set all counters to zero for a particular VLAN filter, all VLAN filters for a particular protocol, or all defined VLAN filters. If you are resetting the counters in a single filter, you can choose the VLAN by specifying the subnet or by selecting the VLAN from a list with the by-name option.

Syntax:
reset-counters
by-name
ip all
ip subnet *subnet address*
ip-multicast all
ip-multicast by-name
ipx all
ipx network *network number*
netbios
sliding-window all
sliding-window by-name
all

Example: reset ipx network 3ff

```
VLAN 'Ethernet B' (IPX Network 0x3FF) counters reset
```

Save Use the **save** command to store the current runtime VLAN configuration

into SRAM. This will overwrite the current SRAM configuration. This command does not affect the runtime behavior of VLANs or reset the timers or counters associated with VLANs.

Syntax: save

Example: save

```
Are you sure you want to save the VLAN configuration to SRAM? [No] y
VLAN configuration saved
```

Show-members

Use the **show-members** command to display all the learned MAC addresses for a particular VLAN that has MAC Address Tracking enabled. Addresses in this list have all transmitted broadcast frames within the configured aging time. The MAC addresses will be displayed along with the associated bridge port and interface and can be sorted by bridge port or increasing MAC address.

Syntax:
show-members
by-name
ip *subnet-address*
ip-multicast
ipx *network-number*
netbios
sliding-window

Example: show-members ip

```
Subnet Address [9.0.0.0]?

        Sort VLAN Members by Port (P) or Mac Address (M) [P]?
        Port Number to Show Membership (0=All) [0]?

        Current Members of Runtime VLAN 'IP 9.x.x.x' (IP Subnet 9.0.0.0):

           Port  1 (Interface  0),  Mac Address: 10.00.5A.00.64.00
           Port  2 (Interface  1),  Mac Address: 10.00.5A.00.65.00
```

Show-vlans

Use the **show-vlans** command to display all the enabled VLANs in which traffic from a particular MAC address has been observed since the last aging timer expiration.

Syntax:

Example: show-vlans

```
Enter Mac Address in Hex: []? 10005A006400

        List of VLANS with Mac Address 10.00.5A.00.64.00:

             VLAN Type    Identifier            VLAN Name
             =========    ==========            =========
          (1) IP           9.0.0.0               IP 9.x.x.x
```

Chapter 29. Using IP

This chapter describes how to configure the Internet Protocol (IP). It includes the following sections:

Basic Configuration Procedures

This section outlines the initial steps required to get the IP protocol up and running. Details about making further configuration changes are covered in other sections of this chapter. Details about individual configuration commands are covered in the command section of this chapter. The following list outlines the initial configuration tasks to bring up IP on the router. After completing these tasks, you must restart the router for the new configuration to take effect.

1. Access the IP configuration environment. (See "Accessing the IP Configuration Environment" on page 307.)
2. Assign IP addresses to network interfaces. (See "Assigning IP Addresses to Network Interfaces".)
3. Enable dynamic routing. (See "Enabling Dynamic Routing" on page 292.)
4. Add static routing information, if necessary. (See "Adding Static Routing Information" on page 293.)
5. Enable ARP subnet routing, if necessary. (See "Enabling ARP Subnet Routing" on page 296.)
6. Set up ARP parameters, if necessary. (See "Setting Up ARP Configuration" on page 296.)
7. Exit the IP configuration process.
8. Restart the router to activate the configuration changes.

Assigning IP Addresses to Network Interfaces

Use the IP configuration **add address** command to assign IP addresses to the network interfaces. The arguments for this command include the interface number (obtained from the `Config>` **list devices** command) and the IP address with its associated address mask.

In the following example, network interface 2 has been assigned the address 128.185.123.22 with the associated address mask 255.255.255.0 (using the third byte for subnetting).

```
IP config> add address 2 128.185.123.22 255.255.255.0
```

Multiple IP addresses can be assigned to a single network interface.

© Copyright IBM Corp. 1996, 1999

Setting the Internal IP Address

This is an IP address that is independent of the state of any interface and is set without reference to any interface. Some IP configurations require it. See the command **set internal-IP-address** on page 343 for more information.

Enabling Dynamic Routing

Use the following procedures to enable dynamic routing on the router. The router software supports OSPF, RIPv1, and RIPv2 for interior gateway protocols (IGPs) as well as BGP, which is an external gateway protocol.

All routing protocols can run simultaneously. However, most routers will probably run only a single routing protocol (one of the IGPs). The OSPF protocol is recommended because of its robustness and the additional IP features (such as equal-cost multipath and variable-length subnets) that it supports.

Setting the Routing Table Size

The routing table size determines the number of entries in the routing table from all sources, including dynamic routing protocols and static routes. The default size is 768 entries.

To change the size of the routing table, use the **set routing table-size** configuration command. Setting the routing table size too small results in routes being discarded. Setting it too large results in inefficient use of memory resources. After operation, use the console **dump** command to view the contents of the table and then adjust the size as necessary, allowing some room for expansion.

Enabling the OSPF Protocol

OSPF configuration is done via its own configuration console (entered via the Config> **protocol ospf** command). To enable OSPF, use the following command:

```
OSPF Config> enable OSPF
```

After enabling the OSPF protocol, you are prompted for size estimates for the OSPF link state database. This gives the router some idea how much memory must be reserved for OSPF. You must supply the following two values that will be used to estimate the size of the OSPF link state database:

- Total number of external routes imported into the OSPF routing domain.
- Total number of OSPF routers in the routing domain.

Enter these values at the following prompts (sample values have been provided):

```
OSPF Config> enable ospf
Estimated # external routes[0]? 200
Estimated # OSPF routers [50]? 60
Maximum LSA size [2048]?
```

Next, configure each IP interface that is to participate in OSPF routing. To configure an IP interface for OSPF, use the following command:

```
OSPF Config> set interface
```

You are prompted to enter a series of operating parameters. Each interface is assigned a cost as well as other OSPF operating parameters.

When running other IP routing protocols besides OSPF, you may want to enable the exchange of routes between OSPF and the other protocols. To do this, use the following command:

```
OSPF Config> enable AS-boundary-routing
```

For more information on the OSPF configuration process, see “Chapter 35. Using OSPF” on page 447.

Enabling the RIP Protocol

This section describes how to initially configure the RIP protocol. When configuring the RIP protocol, you can specify which set of routes the router will advertise and/or accept on each IP interface.

RIP is supported on ATM LAN Emulation network interfaces.

First, enable the RIP protocol with the following command:

```
IP config> enable RIP
```

When RIP is enabled, the following default behavior is established:

- The router includes all network and subnet routes in RIP updates sent out on each of its configured IP interfaces. It does not include default and static routes.
- The router processes all RIP updates received on each of its configured IP interfaces.
- RIP will not override default and static routes.

To change any of the default sending/receiving behaviors, use the following IP configuration commands, which are defined on a per-IP-interface basis.

```
IP config> enable/disable sending net-routes
IP config> enable/disable sending subnet-routes
IP config> enable/disable sending static-routes
IP config> enable/disable sending host-routes
IP config> enable/disable sending default-routes
IP config> enable/disable receiving rip
IP config> enable/disable receiving dynamic nets
IP config> enable/disable receiving dynamic subnets
IP config> enable/disable receiving host-routes
IP config> enable/disable override default
IP config> enable/disable override static-routes
IP config> set originate-rip-default
```

Enabling the BGP Protocol

The BGP protocol is enabled from its own configuration prompt, `BGP Config>` For more information about configuring BGP, refer to the discussion on using and configuring BGP4 in *8371 Interface Configuration and Software User's Guide*.

Adding Static Routing Information

This procedure is necessary only for routing information you cannot obtain from any of the above dynamic routing protocols. Static routing information persists over power failures and is used for routes that never change or cannot be learned dynamically.

The destination of a static route is described by an IP address (*dest-addr*) and an IP address mask (*dest-mask*). The mask indicates the range of IP addresses to which the route applies; for example, a route with IP address 10.0.0.0 and mask

255.0.0.0 applies to IP addresses from 10.0.0.0 through 10.255.255.255. The route to the destination is described by the IP address of the next hop router (*next-hop*) and the cost of forwarding a packet on this route (*cost*).

Longest Match Rule

Because the destination of a route includes the IP address mask, it is possible for more than one route to match a particular IP address; for example, for the IP address 10.1.2.3, a route with IP address 10.0.0.0 and mask 255.0.0.0 and a route with IP address 10.1.0.0 and mask 255.255.0.0 both match. To determine which route to use, the longest match rule is applied. The route with the largest mask is used (in this case the route with IP address 10.1.0.0 and mask 255.255.0.0).

Default, Network, Subnet and Host Routes

Routes can be classified as *default*, *network*, *subnet*, or *host*, according to their destination IP address and mask.

A *default* route has an IP address/mask of 0.0.0.0/0.0.0.0. This route matches all destination IP addresses, but because of the longest match rule, it is used only if there is no other matching route. The following command creates a static default route:

```
IP config> add route
IP destination [ ]? 0.0.0.0
Address mask [255.0.0.0]? 0.0.0.0
Via gateway 1 at [ ]? 192.9.1.4
Cost [1]? 5
Via gateway 2 at [ ]?
IP config>
```

The static default route may also be set by the **set default network-gateway** command; however, this command does not take effect immediately, and it allows you to define only one default static route. The following example creates the same static default route as the above **add route** command:

```
IP config> set default network-gateway
Default gateway [ ]? 192.9.1.4
gateway's cost [1]? 5
IP config>
```

A *network route* has a mask that depends on the value of the route's destination IP address as specified by the IP address classes defined in RFC 791:

IP Address Class	IP Address Range	Network Mask
A	0.0.0.0 - 127.255.255.255	255.0.0.0
B	128.0.0.0 - 191.255.255.255	255.255.0.0
C	192.0.0.0 - 223.255.255.255	255.255.255.0

The **add route**, **change route**, and **delete route** commands use the network mask that corresponds to the destination IP address as the default mask value. The following command creates a static network route:

```
IP config> add route 172.16.0.0
Address mask [255.255.0.0]?
Via gateway 1 at [ ]? 192.9.1.4
Cost [1]? 5
Via gateway 2 at [ ]?
IP config>
```

A static network route may also be set by the **set default subnet-gateway** command; however, this command does not take effect immediately, and it allows

you to define only one static route per destination. The following example creates the same static network route as the above **add route** command:

```
IP config> set default subnet-gateway
For which subnetted network [ ]? 172.16.0.0
Default gateway [ ]? 192.9.1.4
gateway's cost [1]? 5
IP config>
```

A *subnet route* has a mask that is larger than the network mask for the route's destination IP address. The following command creates a static subnet route:

```
IP config> add route 172.16.1.0
Address mask [255.255.0.0]? 255.255.255.0
Via gateway 1 at [ ]? 192.9.1.4
Cost [1]? 5
Via gateway 2 at [ ]?
IP config>
```

A *host route* is a route to a specific IP address; it has a mask of 255.255.255.255. The following command creates a static host route:

```
IP config> add route 172.16.1.2
Address mask [255.255.0.0]? 255.255.255.255
Via gateway 1 at [ ]? 192.9.1.4
Cost [1]? 5
Via gateway 2 at [ ]?
IP config>
```

Interaction Between Static Routing and Dynamic Routing

Routes dynamically learned through the OSPF and RIP protocols can override static routes. For the RIP protocol, you can disable this override behavior. See the RIP section of this chapter concerning the **enable/disable override static-routes** commands.

You can configure both OSPF and RIP to advertise configured static routes over interfaces where these dynamic protocols are enabled.

To configure RIP to advertise static routes, enter the following command at the `IP config>` prompt:

```
IP config> enable sending static-routes ip-interface-address
```

To configure OSPF to advertise static routes, enter the following command at the `OSPF Config>` prompt:

```
OSPF Config>enable as boundary
Use Route Policy [No]?
Import BGP routes [No]?
Import RIP routes [No]?
Import static routes [No]? yes
Import direct routes [No]?
Import subnet routes [Yes]?

OSPF Config>enable as boundary
Import static routes [yes]?
```

Nexthop Awareness

Nexthop Awareness allows the router to sense whether a neighboring router is up or down. When this option is enabled, the router makes a more accurate determination of whether a static route that uses the neighboring router as its next hop will function. It also allows the router to determine over which network interface a static route's next hop can be reached when that next hop is in an IP subnet that is defined on multiple network interfaces.

To enable Nexthop Awareness on a particular IP interface, enter the following command at the IP configuration prompt:

```
IP config> enable nexthop-awareness ip-interface-address
```

To disable Nexthop Awareness on a particular IP interface, enter the following command at the IP configuration prompt:

```
IP config> disable nexthop-awareness ip-interface-address
```

Nexthop Awareness is supported only on frame relay networks on which the neighboring routers support inverse ARP.

Setting Up ARP Configuration

The Address Resolution Protocol (ARP) is used to map protocol addresses to hardware addresses before a packet is forwarded by the router. ARP is always active on the router, so you do not need to do any additional configuration to enable it with its default characteristics. However, if you need to alter any ARP configuration parameters (such as **enable auto-refresh** or **set refresh-timer**, which changes the default refresh timer), or if you need to add, change, or delete permanent address mappings, see “Chapter 33. Using ARP” on page 435.

If LAN Emulation is configured on an interface, the defaults apply. You can effectively use the ARP protocol without any changes.

Enabling ARP Subnet Routing

If there are hosts on attached subnetted networks that do not support IP subnetting, use Address Resolution Protocol (ARP) subnet routing (described in RFC 1027). When the router is configured for ARP subnet routing, it will reply by proxy to ARP requests for destination (that is, off the LAN if the router is itself the best route to the destination, and the destination is in the same natural network as the source). For correct operation, all routers attached to a LAN containing subnetting-ignorant hosts should be configured for ARP subnet routing.

To enable ARP subnet routing, use the following command:

```
IP config> enable arp-subnet-routing
```

Enabling ARP Network Routing

Some IP hosts use ARP for all destinations, whether or not the destination is in the same natural network as the source. For these hosts, ARP subnet routing is not enough, and the router can be configured to reply by proxy to any ARP request as long as the destination is reachable through the router and the destination is not on the same local network segment as the source.

To enable ARP network routing, use the following command:

```
IP config> enable arp-network-routing
```

IP Filtering

Filtering allows you to specify certain criteria that the router uses to control packet forwarding. The following main types of filtering are provided to help you achieve your security and administrative goals:

- Access control

- Route filtering

Access Control

Access control allows the IP router to control the processing of individual packets based on source and destination IP addresses, IP protocol number, and by destination port number for the TCP and UDP protocols. This can control access to particular sets of IP hosts and services.

You can define access controls by configuring access control lists. One global list and two lists per interface can be specified. The global list applies to the router as a whole. Interface lists, also known as packet filters, are assigned names and apply only to the designated interface. For each interface, one list applies to incoming packets, and the other applies to outgoing packets. The lists are applied independently of each other. A packet might *pass* an incoming interface list, and be *dropped* by the global list.

Figure 23 illustrates the series of access control lists through which a packet must pass before being forwarded.

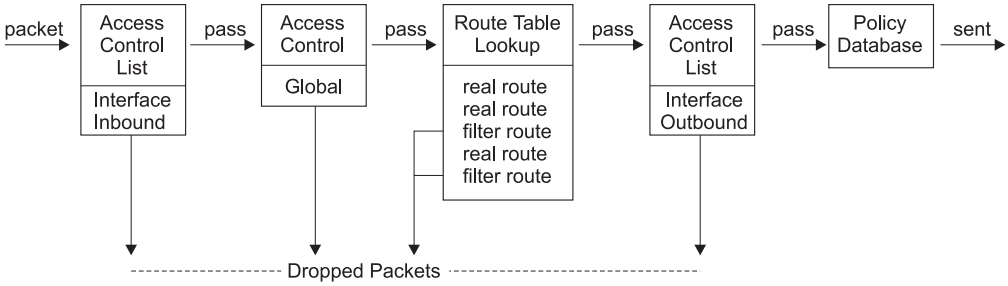

Figure 23. Access Control Lists in the Packet Forwarding Path

Access Control Rules

Each access control list consists of one or more access control rules that set the filtering criteria. Some access control rules define the global filters that affect all the interfaces on the router and others define the interface-specific access control lists (also called packet filters). The global access control rules are configured using the **add access** command at the `IP config>` prompt. The packet filters are set using two commands at the `IP config>` prompt: the **add packet-filter** command to define the filter and the **update packet-filter** command to configure it.

As IP packets flow through the router, IP packet fields are compared to the access control rules. A packet matches a rule if every specified field in the rule matches a corresponding field in the packet. If a packet matches a rule, and the rule filter type is inclusive, the packet *passes*. If the rule filter type is exclusive, the packet is *dropped* and is not processed any further by the router. If no rules match after going through the entire list, the packet is also dropped.

When defining records in access control lists, it is important to remember the following information:

- The order of records in a list is important. Configuration commands are provided to change the order of records in a list.
- For every list that includes at least one access control rule, an inclusive rule must exist for any packets that do not match any of the access control rules to pass

the list. One method of allowing all packets that do not match any of the specified rules to pass is to include the following wildcard rule as the last rule in the list:

```
IP config> add access-control
Enter type [E]? i
```

Enabling Access Control

IP Access Control (including global and interface access control) is enabled with the **set access-control on** command and disabled with the **set access-control off** command. You can use the **enable packet-filter** and the **disable packet-filter** commands to enable and disable specific packet filters when IP access control is enabled.

If IP access control is enabled, you must be careful with packets that the router originates and receives. Be sure not to filter out the RIP or OSPF packets being sent or received by the router. The easiest way to do this is to add a wildcard inclusive rule as the last in the access control list. Alternatively, you can add specific rules for RIP and OSPF, perhaps with restrictive addresses and masks. Note that some OSPF packets are sent to the Class D multicast addresses 224.0.0.5 and 224.0.0.6, which is important if address checking is being done for routing protocols. See the **add** command for more information on access control.

Defining the Global Access Control List

The global access control list is defined when rules are added at the `IP config>` prompt:

```
IP config>  add access-control...
```

Global access control rules can be listed, moved, or deleted using the **list**, **move**, or **delete** commands. See these commands for further information.

Defining Packet Filters

To define packet filters, which are interface-specific, use the **add packet-filter** command at the `IP config>` prompt. The router prompts you for the filter name, direction (input or output), and the interface number to which it applies.

```
IP config> add packet filter
Packet-filter name [ ]? test
Filter incoming or outgoing traffic? [IN]? in
Which interface is this filter for [0]? 1
```

You can use the **list packet-filter** command to list all interface-specific access control lists configured in the router.

Setting Up Access Control Rules for Packet Filters

You must define access control rules for each defined list (packet filter). Otherwise, defined packet filters will have no effect on incoming or outgoing traffic. Use the **update packet-filter** command at the `IP config>` prompt to define access control rules. The router first prompts you for the name of the packet filter that you want to update. The `IP config>` prompt then changes to `Packet-filter 'name' Config>` where ‘name’ is the list name that you provide.

```
IP config>  update packet-filter
Packet-filter name [ ]?  test
Packet-filter 'test' Config>
```

From this prompt, you can issue **add**, **list**, **move**, and **delete** commands. These commands are similar to those used to modify the global access control list.

Parameters for Access Control Rules

Access control rules consist of multiple parameters. Some parameters can be specified in all access control rules, while others can be specified only in the rules for packet filters. The following parameters can be specified in all access control rules:

- Type (inclusive, exclusive)
- IP source address and mask
- IP destination address and mask
- IP protocol number range
- TCP/UDP port number range
- Precedence and TOS filtering support
- Policy-based routing (selecting the next-hop gateway)

The following parameter is for packet filters only:

- Packet filter name

Type: The type designation of an access control rule defines what it does to packets that match it. An *exclusive* (E) rule discards packets. An *inclusive* (I) rule allows packets to be processed further by the router.

IP Source and Destination Addresses and Masks: Each rule has an IP address and mask pair for both the IP source and destination addresses. When an IP packet is compared to an access control rule, the IP address in the packet is ANDed with the mask in the rule, and the result compared with the address in the rule. For example, a source address of 26.0.0.0 with a mask of 255.0.0.0 in an access control rule will match any IP source address with 26 in the first byte. A destination address of 192.67.67.20 and a mask of 255.255.255.255 will match only IP destination host address 192.67.67.20. An address of 0.0.0.0 with mask 0.0.0.0 is a wildcard that matches any IP address.

IP Protocol Number Range: Each record can also have an IP protocol number range. This range is compared to the protocol byte in the IP header; a protocol value within the range specified by the access control rule will match (including the first and last numbers of the range). If you specify a range of 0 to 255, any protocol will match. Commonly used protocol numbers are 1 (ICMP), 6 (TCP), 17 (UDP), and 89 (OSPF).

TCP/UDP Port Number Range: TCP/UDP port number ranges can also be specified in an access control rule. This range is compared to the port number field in the TCP or UDP header of the IP packet; a port number value within the specified range (inclusive) will match. This field is ignored for IP packets that are not TCP or UDP packets. If you specify a range of 0 to 65535, any port number will match. Commonly used port numbers are 21 (FTP), 23 (Telnet), 25 (SMTP), 513 (rlogin) and 520 (RIP). See RFC 1700 (Assigned Numbers) for a list of IP protocol and port numbers.

Precedence and TOS Filtering Support: The router that supports TOS has identified certain routes that provide the requested levels of service. The router sends packets over the routes according to the setting of their TOS bits.

TOS in IP is not a guarantee of any particular type of service, but a request to the router to provide service of the type requested. For example, a packet with a TOS field requiring maximum throughput can be sent over several hops that have

different bandwidths. It will get normal service - no special treatment - if it should pass over a hop managed by a router that does not support TOS. See the **add access-controls** command on page 309 for descriptions of these parameters.

Parameters for TOS-Based Routing Support: To enable the router to interpret TOS bits and route packets according to those bits, you create an access control rule from which the router will receive TOS packets for filtering and Type of Service routing. This access control rule applies to all the interfaces on the router. The following parameters are used to define the TOS bits that the router will compare:

- Range-start value for the TOS byte bits
- Range-end value for the TOS byte bits
- Filter mask to determine which bits in the TOS byte are included in the range

Modification of the TOS Bits: To enable the router to modify the TOS bits of incoming packets, you create a global access control rule from which the router will receive TOS packets that are to be modified. Modifying the value of the TOS bits is a separate activity from interpreting them and routing the packet. If both interpretation and modification are configured, the modification will be done after the interpretation. The following parameters are used to define the TOS bits to be modified:

- A new value for the TOS bits
- A modification mask to determine which bits in the TOS byte are to be changed

Policy-Based Routing (Selecting the Next-Hop Gateway): You can filter inbound packets to direct them to a manually selected next hop gateway address (known as policy-based routing). To do this, create an inclusive inbound access control rule either globally, for the router, or for a particular interface, and provide the following parameters:

- Whether to use policy-based routing
- The IP address of the next hop gateway
- Whether or not to send the packet using the normal routing table if the next hop is unavailable

Examples: The following example allows any host to send packets to the SMTP TCP socket on 192.67.67.20.

```
add access-control inclusive 0.0.0.0 0.0.0.0 192.67.67.20 255.255.255.255 6 6 25 25
```

The next example prevents any host on subnet 1 of Class B network 150.150.0.0 from sending packets to hosts on subnet 2 of Class B network 150.150.0.0 (assuming a 1-byte subnet mask).

```
add access-control exclusive 150.150.1.0 255.255.255.0 150.150.2.0 255.255.255.0 0 255 0 65535
```

This command allows the router to send and receive all RIP packets.

```
add access-control inclusive 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 17 17 520 520
```

This example shows how to create a global access control rule. Values are entered to enable the interpretation of TOS bits of packets arriving from IP address 9.1.2.3 and to change the values of these bits before sending the packets. See “Add” on page 309 for an explanation of the meaning of the parameters that create TOS filtering and policy-based routing.

```
IP config> add access-control
Enter type [E]? i
Internet source [0.0.0.0]? 9.1.2.3
Source mask [255.255.255.255]?
Internet destination [0.0.0.0]?
```

```
Destination mask [0.0.0.0]?
Enter starting protocol number ([0] for all protocols) [0]?
Enter starting DESTINATION port number ([0] for all ports) [0]?
Enter starting SOURCE port number ([0] for all ports) [0]?
Filter on ICMP Type ([-1] for all types) [-1]?
TOS/Precedence filter mask (00-FF - [0] for none) [0]? e0
TOS/Precedence start value (00-FF) [0]?
TOS/Precedence end value [0]?
TOS/Precedence modification mask (00-FF - [0] for none) [0]? 1f
New TOS/Precedence value (00-FF) [0]? 08
Use policy-based routing? [No]: y
Next hop gateway address [ ]? 9.2.160.1
Use default route if next hop gateway unreachable? [Yes]:
Enable Logging (Yes or [No]):
```

Route Filtering Without Policies

Route filtering impacts packet forwarding by influencing the content of the routing table. In general, route filtering is more efficient but less flexible than access control. Filtering based on packet fields other than the destination IP address can be done using access control, described above.

The following methods are used in this router to influence the content of the routing table.

- Filter routes
- RIP input filters
- Route table filtering

Defining Filter Routes

You can designate an IP destination to be inserted in the routing table as a *filter route*. IP packets will not be forwarded to these destinations, and routing information concerning them will not be advertised. Filter routes are ***not*** recommended when OSPF is used in your network; OSPF-learned internal routes will override filtered routes in the routing table.

To configure a filter route, enter the following command at the `IP config>` prompt:

```
IP config> add filter dest-IP-address address-mask
```

Filter routes will be listed as an entry with the type *fltr* when the **dump** command is used to view the IP routing table.

Note: If a more specific route is available, packets will be forwarded. For example, if a filter route is defined for network 9.0.0.0 (mask 255.0.0.0), but a route is learned for a subnet of the network (for example 9.1.0.0, mask 255.255.0.0), then packets will be forwarded to subnet 9.1.0.0 but not to other subnets of that network.

Defining RIP Input Filters

When RIP is used as the dynamic routing protocol, you can configure certain interfaces to ignore routes in RIP updates.

The following command results in ignoring all RIP updates received on an interface:

```
IP config> disable receiving rip ip-interface-address
```

The following commands result in ignoring certain types of routes received on an interface:

```
IP config> disable receiving dynamic nets ip-interface-address
IP config> disable receiving dynamic subnets ip-interface-address
IP config> disable receiving dynamic host ip-interface-address
```

If more granular filtering of RIP routes is required, the route policies that are described in the following command can be utilized:

```
IP config> add accept-rip-route ip-network/subnet/host
```

Defining Route Table Filtering

When route table filtering is enabled and route filters are defined, checking is performed before adding routes to the IP routing table. If the route to be added matches on an inclusive route filter, it will be added to the IP route table. If it matches on an exclusive route filter, it will not be added to the IP route table. Direct and static routes will never be filtered.

This function can be used to prevent routes from being added to the IP route table in situations where the network administrator does not want all routes advertised by routing protocols to be available. This function could be used in a service provider environment to prevent customers from having access to each other's networks.

Configuring the BOOTP/DHCP Forwarding Process

BOOTP (documented in RFC 951 and RFC 1542) is a bootstrap protocol used by a diskless workstation to learn its IP address, the location of its boot file, and the boot server name. Dynamic Host Configuration Protocol (DHCP), documented in RFC 2131, is used to allocate reusable network addresses and host-specific configuration parameters from a server.

The following terms are useful when discussing the BOOTP/DHCP forwarding process:

- *Client* - the workstation requiring BOOTP/DHCP services.
- *Servers* - the boot host (with UNIX daemon bootpd, DOS version available from FTP software, or OS/2) or other DHCP/BOOTP server that is providing these services.
- *BOOTP relay agent* or *BOOTP forwarder* - a device that forwards requests/replies exchanged by the Client and Server. This router can provide the relay agent function.

The following steps outline an example of the BOOTP forwarding process. (DHCP exchanges proceed in a similar way):

1. The Client copies its Ethernet address (or appropriate MAC address) into a BOOTP packet and broadcasts it onto the local LAN. BOOTP is running on top of UDP.
2. The local BOOTP relay agent receives the packet and checks to see if the packet is well formatted and that the maximum number of application hops has not expired. It also checks to see if the client has been trying long enough.

   Note: If multiple hops are required before reaching the BOOTP agent, the packet is routed normally via IP. All other routers would not examine the packet to determine whether it is a BOOTP packet.
3. The local BOOTP agent forwards a separate BOOTP request to each of its added servers. The BOOTP request is the same as the one that was initially sent by the client except that it has a new IP header with the relay agent's IP address copied into the body of the BOOTP request.

4. The server receives the request and looks up the client's hardware (for example, 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 to the BOOTP relay agent.
5. The BOOTP relay agent receives the reply and makes an entry in its ARP table for the client and then forwards the reply to the client.
6. The client then continues to boot using TFTP, using the information in the BOOTP reply packet.

Enabling/Disabling BOOTP Forwarding

To enable or disable BOOTP forwarding on the router, enter the following command at the IP configuration prompt. (Enable BOOTP Forwarding to allow the router to forward BOOTP and/or DHCP requests and replies between Clients and Servers on different segments of your network.)

```
IP config> enable/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 Server. A typical value for this parameter is 1.
- Number of seconds you want the Client to retry before the BOOTP request is forwarded. *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 router replicates the packet.

Adding a BOOTP/DHCP Server

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

```
IP config> add bootp-server server-IP-address
```

Multiple servers can be configured. In addition, if only the network number of the server is known or if multiple servers reside on the same network segment, a broadcast address can be configured for the server.

Configuring Virtual Router Redundancy Protocol (VRRP)

The use of a statically configured default route is popular for host IP configurations. It minimizes configuration and processing overhead and is supported by virtually every IP implementation. This mode of operation is likely where dynamic host configuration protocols are deployed that typically provide configuration for an end-host IP address and default gateway. However, this creates a single point of failure. Loss of the default router results in a catastrophic event, isolating all end-hosts that are unable to detect any alternate path that may be available.

The Virtual Router Redundancy Protocol (VRRP) is designed to eliminate the single point of failure inherent in the static default routed environment. VRRP specifies an election protocol that dynamically allows a set of routers to back up each other. The

VRRP router controlling one or more IP addresses is called the master router, and forwards packets sent to these IP addresses. The election process provides dynamic fail-over in the forwarding responsibility should the master become unavailable. Any of the IP addresses on a virtual router can then be used as the default first hop router by end-hosts. The advantage gained from using the VRRP is a higher availability default path without requiring configuration of dynamic routing or router discovery protocols on every end-host.

In order to use and configure VRRP you must first define a Virtual Router ID (VRID) on each LAN segment running VRRP. The VRID is a number in the range of 1 to 255. This VRID identifies the routers that will back one another up. Therefore, all VRRP routers that are backups for one another must have the same VRID. For each VRRP segment, one router called the master router owns the default IP address configured for hosts on the LAN segment. As long as the master is available, it responds to ARP requests for that address and forwards packets. One of the backup routers takes the place of the master router if the master router becomes unavailable. When a backup router takes over, it becomes accessible at the default IP address so that the hosts now use it as the master router.

The VRID represents a unicast or multicast virtual MAC address. You can configure the backup routers with a virtual MAC address or configure each VRRP router to use its own unique burned-in hardware MAC address. If you use the multicast option, you cannot use the hardware MAC address. If you use the hardware MAC address, the hosts that communicate with the VRRP router must support gratuitous ARPs. Using the hardware MAC address can provide improved performance in your network.

The following is an example of a very simple VRRP topology. In this example, the virtual MAC address is used. If the hardware MAC address were used, the master router and the backup router would each use its own hardware MAC address.

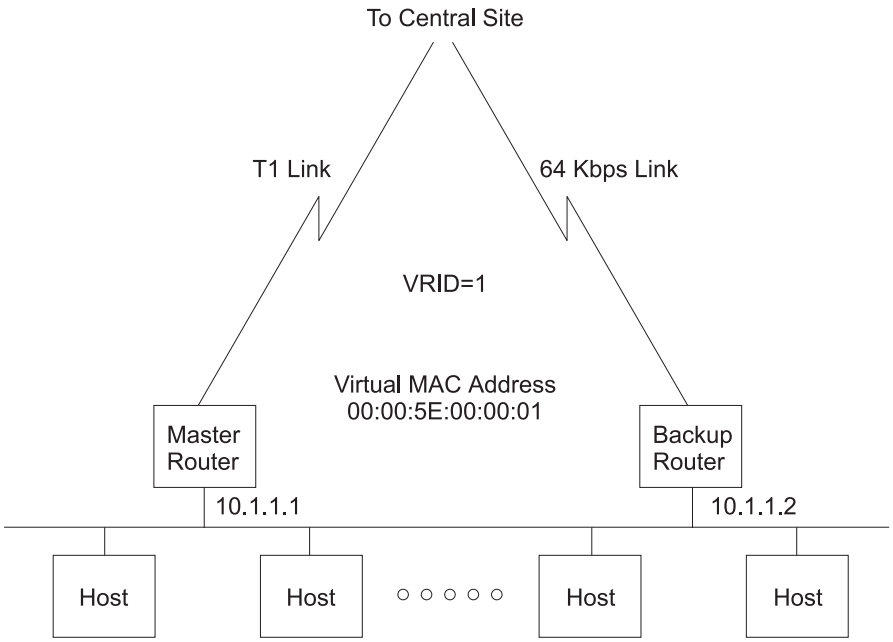

Figure 24. Ethernet LAN with subnet 10.1.1.0/255.255.255.0 All Host Configured with Default Gateway 10.1.1.1

1. All hosts are configured with default gateway of 10.1.1.1
2. The master router will answer to all ARP requests for 10.1.1.1 with the virtual MAC address of 00:00:5E:00:00:01.
3. The master router will forward packets addressed to the virtual MAC address.
4. If the master router is unavailable, the backup determines this via the absence of VRRP advertisements and will commence receiving packets addressed to the virtual MAC address. The backup will also answer to ARP requests for 10.1.1.1.

A complicated topology would be one where there are multiple VRRP routers and the desire is to balance the load between the routers but still have complete backup capability. In this case 2 VRIDs would need to be defined and each router would the master for one and the backup for the other. This illustration follows:

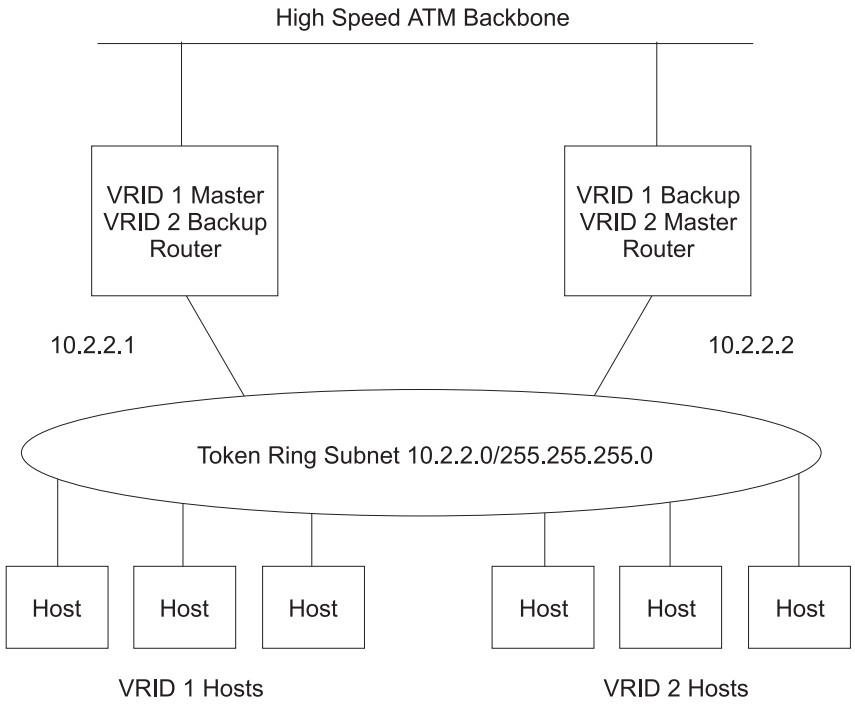

Figure 25. Multiple VRRP Routers

1. All VRID 1 hosts will be configured with a default gateway address of 10.2.2.1.
2. All VRID 2 hosts will be configured with a default gateway address of 10.2.2.2.
3. The VRID 1 master router will respond to ARP requests for address 10.2.2.1 with the virtual MAC address C0:00:00:10:00:00. It will also receive and forward packets addressed to virtual MAC address C0:00:00:10:00:00.
4. The VRID 2 master router will respond to ARP requests for address 10.2.2.2 with the virtual MAC address C0:00:00:20:00:00. It will also receive and forward packets addressed to virtual MAC address C0:00:00:20:00:00.
5. If either router becomes unavailable, the other will take over.
6. If a router does not become unavailable but loses it's outside connectivity, it will re-direct traffic through the other with ICMP redirects (this assumes the 2 routers are exchanging routes via routing protocol such as RIP or OSPF).

Configuring the Redundant Default IP Gateway

This section outlines the steps used to configure redundant default IP gateways on ELANs. Configuration of a redundant gateway allows end stations with manually configured default gateways to continue passing traffic to other subnets after their primary gateway goes down.

To configure a device with a primary gateway or backup gateway:

1. Determine the IP address end stations use as the default gateway.
2. Determine a MAC address not used by any interfaces on the ELAN. To determine which MAC addresses are used, see “Database List” in the “Monitoring LAN Emulation Services” chapter of *8371 Interface Configuration and Software User’s Guide*.
3. Select a device to have the primary gateway. This device must have a LEC interface on the ELAN of the end station.
4. Select a device or set of devices to have the backup gateway. This device or set of devices must have a LEC interface on the ELAN of the end station.
5. Config a redundant gateway on each device using the “Add” option for IP.

Note: The primary gateway and the backup gateway must have the same MAC address

Chapter 30. Configuring and Monitoring IP

This chapter describes the IP configuring and monitoring commands. It includes the following sections:

- "Accessing the IP Configuration Environment"
- "IP Configuration Commands"
- "IP Monitoring Commands" on page 355
- "Route Filter Policy Configuration" on page 349
- "Accessing the IP Monitoring Environment" on page 355

Accessing the IP Configuration Environment

To access the IP configuration environment, enter the following command at the `Config>` prompt:

```
Config> Protocol IP
Internet protocol user configuration
IP config>
```

IP Configuration Commands

This section describes the IP configuration commands. These commands allow you to modify the IP protocol behavior to meet your specific requirements. Some amount of configuration is necessary to produce a fully functional IP router. Enter IP configuration commands at the `IP config>` prompt.

Table 56. IP Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds to the IP configuration information. Interface addresses can be added, along with access controls, filters, and packet-filters.
Change	Modifies information that was originally entered with the **add** command.
Delete	Deletes IP configuration information that had been entered with the **add** command.
Disable	Disables certain IP features that have been turned on by the **enable** command.
Enable	Enables IP features such as ARP subnet routing, UDP Forwarding, originate default, directed broadcasts, BOOTP, the various RIP flags controlling the sending and receiving of RIP information, diffserv, and route-table-filtering.
List	Displays IP configuration items.
Move	Changes the order of access control records.
Set	Establishes IP configuration modes such as the use of access control and the format of broadcast addresses. Also sets IP parameters such as TTL (time-to-live) of packets originated by the router, the size of the IP routing table, and RIP interface metrics.
Update	Used to assign access control entries to packet filters.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

© Copyright IBM Corp. 1996, 1999

Response to IP Configuration Commands

This topic enables you to determine which IP configuration (Talk 6) commands become effective immediately and which commands remain pending until you issue the Talk 5 **reset ip** command to a router. Table 57 lists both categories of commands. Any commands that are not listed in the table remain pending until you issue a **reload** command.

Table 57. IP Configuration Command Response

Effective Immediately	Effective at Reset
add route	add accept-rip-route ...
change route	add access-control ...
delete route	add address
disable icmp-redirect	add bootp-server
enable icmp-redirect	add packet-filter
set ttl	add udp-destination
	add vrid ...
	add vr-address
	change access-control ...
	change address ...
	delete accept-rip-route ...
	delete access-control ...
	delete address ...
	delete bootp-server
	delete packet-filter
	delete udp-destination
	delete vrid ...
	delete vr-address ...
	disable bootp-forwarding
	disable directed-broadcast
	disable echo-reply
	disable fragment-offset-check
	disable icmp-redirect
	disable nexthop-awareness ...
	disable override default/static-routes...
	disable packet-filter
	disable receiving ...
	disable record-route
	disable rip
	disable rip2
	disable same-subnet
	disable sending ...
	disable source-addr-verification
	disable source-routing
	disable timestamp
	disable trace
	disable udp-forwarding
	disable vrrp ...
	enable bootp-forwarding
	enable directed-broadcast
	enable echo-reply
	enable fragment-offset-check ...
	enable icmp-redirect
	enable nexthop-awareness
	enable override ...
	enable packet-filter

Table 57. IP Configuration Command Response (continued)

Effective Immediately	Effective at Reset
	enable receiving ...
	enable record-route
	enable rip
	enable rip2
	enable same-subnet
	enable sending ...
	enable source-address-verification
	enable source-routing
	enable timestamp
	enable trace
	enable udp-forwarding
	enable vrrp ...
	move access-control ...
	set access-control ...
	set access-control log-facility
	set broadcast-address ...
	set originate-rip-default
	set rip-in-metric
	set rip-out-metric
	set tag ...
	set ttl
	update packet-filter ...

Add

Use the **add** command to add IP information to your configuration.

Syntax:

add accept-rip-route . . .
access-control . . .
address . . .
bootp-server
distributed default gateway
filter . . .
packet-filter
redundant default gateway
route . . .
route-table-filter
vrid . . .
vr-address . . .

accept-rip-route *IP-network/subnet*
Allows an interface to accept a RIP route when input RIP filtering is enabled for an interface. You can print the list of networks and subnets that have already been entered using the **list rip** command. You can enable the input filtering of RIP routes on a per-IP-interface basis. This is done separately for network-level routes (for example, a route to 10.0.0.0) for subnet-level routes (for example, a route to 128.185.0.0), and for host-level routes (for

example 128.185.123.28). To enable input filtering of routes on an IP interface, use the **disable receiving dynamic nets** or **disable receiving dynamic subnets** or **disable receiving dynamic hosts** commands.

IP network/subnet

Valid Values: any valid IP address

Default Value: none

Example:

```
add accept-rip-route

Network number [0.0.0.0]? 10.0.0.0
```

access-control *type IP-source source-mask IP-dest dest-mask first-protocol last-protocol [first-port last-port]*

Adds an access control record to the end of the global access control list. This allows you to describe a class of packets to forward or drop, depending on the type of the record. The length and order of the IP access control list can affect the performance of the IP forwarder. Each record must be assigned the following: type, IP source, source-mask, IP destination, and destination-mask fields. The type must either be inclusive or exclusive. The *IP-source* and *IP-dest* fields are in the form of IP addresses in dotted decimal notation. Optionally, you can specify an IP protocol number range with the *first-protocol* and *last-protocol* fields, which are an inclusive range of IP protocols that match this entry. You can also specify a TCP or UDP port number or port number range that matches an entry, where “port number range” is an inclusive range of TCP and UDP ports that matches this entry. Specify TCP or UDP in the protocol fields, then specify the port number range in the first-port and last-port fields.

type Indicates whether packets are sent or dropped for a specific address or set of addresses.

Specify *Include* to cause the router to receive a packet and to forward it if it matches criteria in the remaining arguments.

Specify *Exclude* to cause the router to discard the packets.

IP-source

Valid Values: any valid IP address

Default Value: none

source-mask

Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: none

IP-dest

Valid Values: any valid IP address

Default Value: none

dest-mask

Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: none

first-protocol

The lower boundary of a range of IP protocol numbers.

Some commonly used protocol numbers are:
1 for ICMP
6 for TCP
17 for UDP
89 for OSPF

Valid Values: 0 to 255

Default Value: 0

last-protocol
The upper boundary of a range of IP protocol numbers.

Some commonly used protocol numbers are:
1 for ICMP
6 for TCP
17 for UDP
89 for OSPF

Valid Values: 0 to 255

Default Value: 255

first-port
The lower boundary of an IP TCP/UDP port number range.

Some commonly used port numbers are:
21 for FTP
23 for Telnet
25 for SMTP
513 for rlogin
520 for RIP

Valid Values: a port number in the range of 0 - 65535

Default Value: 0

last-port
The upper boundary of an IP TCP/UDP port number range.

Some commonly used port numbers are:
21 for FTP
23 for Telnet
25 for SMTP
513 for rlogin
520 for RIP

Valid Values: a port number in the range 0 to 65535

Default Value: 65535

Example: `add access-control inclusive`

```
Internet source [0.0.0.0]?
Source mask [255.255.255.255]?
Internet destination [0.0.0.0]?
Destination mask [255.255.255.255]?
Enter starting protocol number ([CR] for all) [-1]?
IP config>
```

address *interface-number IP-address address-mask*
Assigns an IP address to one of the router's hardware network interfaces. A hardware network interface will not receive or transmit IP packets until it has at least one IP address. You must specify an IP address together with

its subnet mask. For example, if the address is on a class B network, using the third byte for subnetting, the mask would be 255.255.255.0. Use the **list devices** command to obtain the appropriate command interface-number.

You must specify an IP address together with its subnet mask. For example, if the address is on a class B network, using the third byte for subnetting, the mask would be 255.255.255.0. Use the **List Devices** option to obtain the appropriate option interface-number.

interface-number

Valid Values: any defined interface number

Default Value: none

ip-address

Valid Values:

The class A range is 1.0.0.1 through 126.255.255.254
The class B range is 128.0.0.1 through 191.255.255.254
The class C range is 192.0.0.1 through 223.255.255.254

Default Value: none

address mask

Valid Values: 0.0.0.0 - 255.255.255.255

Default Value: none

Example: `add address 0 128.185.123.22 255.255.255.0`

bootp-server *server-IP-address*

Adds a BOOTP/DHCP server to the list of servers to which the router will forward BOOTP/DHCP requests. See “Configuring the BOOTP/DHCP Forwarding Process” on page 302 for more information.

server-IP-address

Valid Values: any valid Bootp server IP address

Default Value: none

Example: `add bootp-server 128.185.123.22`

distributed default gateway*interface-number gateway-IP-address address-mask MAC-address primary-gateway*

Adds a Distributed Gateway IP address to your configuration.

interface-number

Specifies the net number of LEC interfaces on the ELAN.

Valid Values: net numbers of LEC interfaces

Default Value: none

gateway-IP-address

Specifies the Default Gateway of the end station.

Valid Values: IP addresses used as default gateways

Default Value: 0.0.0.0

address-mask

Specifies the mask of the IP address.

Valid Values: any valid IP net mask

Default Value: 0.0.0.0

MAC-address

Valid Values: any valid MAC address not used by other interfaces on the ELAN

Default Value: 00.00.00.00.00.00

primary-gateway

Specifies whether the gateway is used as the primary or as the backup gateway.

This query asks whether the gateway on this device is the primary gateway active during the normal operation of the network, or the backup gateway that is active when the LEC interface containing the primary gateway is not operational. Answering **Yes** configures a primary gateway. There should be only one primary gateway per ELAN.

Valid Values Yes or No

Default Value: No

Example: `add distributed`

```
Which net is this distributed gateway for [0]? 1
IP address of gateway [0.0.0.0]? 9.67.205.1
Address mask [255.255.0.0]? 255.255.240.0
MAC address [00.00.00.00.00.00.]? 00.00.00.00.00.BA
Is this the primary gateway [No]? Yes or No
```

filter *dest-IP-address address-mask*

Designates an IP destination to be filtered. IP packets will not be forwarded to filtered destinations, nor will routing information be disseminated concerning such destinations. Packets to filtered destinations are simply discarded. You must specify a filtered destination as an IP address with its subnet mask. For example, to filter a subnet of a class B network, using the third byte for subnetting, the mask would be 255.255.255.0. Using the filter mechanism is more efficient than IP access controls, although not as flexible. Filters also affect the operation of the IP routing protocols, unlike access controls. Filtered networks/subnets are overridden if learned using the OSPF routing protocol.

The effect of this command is immediate; you do not have to reboot the router for it to take effect.

dest-IP-address

Valid Values: any valid IP address

Default Value: none

address mask.

Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: 0.0.0.0

Example: `add filter 127.0.0.0 255.0.0.0`

packet-filter *filter-name type interface-number*

Defines a packet filter record within the router configuration.

filter-name

Valid Values: any 16-character name.

You can include dashes (-) and underscores (_) in the name.

Default Value: none

type *IN* filters incoming traffic.

OUT filters outgoing traffic.

interface-number

Valid Values: any defined interface

Default Value: none

Example: `add packet-filter`

```
Packet-filter name [ ]? filt-1-0
Filter incoming or outgoing traffic? [IN]?
Which interface is this filter for [0]? 1
```

redundant default gateway *interface-number gateway-IP-address address-mask MAC-address primary-gateway*

Adds a Redundant Default Gateway IP address to your configuration.

interface-number

Specifies the net number of LEC interfaces on the ELAN.

Valid Values: net numbers of LEC interfaces

Default Value: none

gateway-IP-address

Specifies the Default Gateway of the end station.

Valid Values: IP addresses used as default gateways

Default Value: 0.0.0.0

address-mask

Specifies the mask of the IP address.

Valid Values: any valid IP net mask

Default Value: 0.0.0.0

MAC-address

Note: The primary gateway and the backup gateway must have the same MAC address

Valid Values: any valid MAC address not used by other interfaces on the ELAN

Default Value: 00.00.00.00.00.00

primary-gateway

Specifies whether the gateway is used as the primary or as the backup gateway.

This query asks whether the gateway on this device is the primary gateway active during the normal operation of the network, or the backup gateway that is active when the LEC interface containing the primary gateway is not operational. Answering **Yes** configures a primary gateway. There should be only one primary gateway per ELAN.

Valid Values Yes or No

Default Value: No

Example: `add redundant`

```
Which net is this redundant gateway for [0]? 1
IP address of gateway [0.0.0.0]? 9.67.205.1
Address mask [255.255.0.0]? 255.255.240.0
MAC address [00.00.00.00.00.00.]? 00.00.00.00.00.BA
Is this the primary gateway [No]? Yes or No
```

route *dest-addr dest-mask next-hop1 cost1 [next-hop2 cost2 [next-hop3 cost3 [next-hop4 cost4]]]*

Adds 1 to 4 static routes to the device's IP configuration. When dynamic routing information is not available for a particular destination, static routes are used.

The destination is specified by an IP address (*dest-addr*) together with an address mask (*dest-mask*). If the destination IP address is a network address, then the dest-mask must be a network mask. If the destination IP address is a subnet address, then the dest-mask must be a subnet mask. Finally, if the destination IP address is a host address, then the dest-mask must be a host mask (which means that the only valid value is 255.255.255.255). The dest-mask must be accurate; if it is not, the static route will not be accepted.

The route to the destination is specified by the IP address of the next hop *(next-hop)*, and the cost *(cost)* of routing the packet to the destination. The next hop must be on the same (sub)net as one of the router's directly connected interfaces. Static routes are always overridden by routes learned through OSPF, but, by default, routes learned through RIP do not override static routes. However, you can enable or disable routes learned through RIP to override static routes by using the **enable override static-routes** or **disable override static-routes** commands. This command takes effect immediately; you do not have to reboot the router.

dest-addr

Valid Values: any valid IP address

Default Value: none

dest-mask

Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: none

next-hop1, next-hop2, next-hop3, next-hop4

Valid Values: any valid IP address

Default Value: none

cost1, cost2, cost3, cost4

Valid Values: an integer in the range 0 to 255

Default Value: 1

Example:

```
IP config> add route
IP destination []? 1.1.0.0
Address mask [255.0.0.0]? 255.255.0.0
Via gateway 1 at []? 10.1.1.1
Cost [1]? 1
Via gateway 2 at []?
IP config> add route 1.1.0.0 255.255.0.0
Via gateway 2 at []? 20.1.1.1
Cost [1]? 2
Via gateway 3 at []? 30.1.1.1
Cost [1]? 3
Via gateway 4 at []?
IP config> add route 2.2.0.0 255.255.0.0 10.2.2.2 1 20.2.2.2 2
IP config> list routes

route to 1.1.0.0        ,255.255.0.0     via 10.1.1.1        cost 1
                                         via 20.1.1.1        cost 2
```

```
                                                  via 30.1.1.1        cost 3
                route to 2.2.0.0       ,255.255.0.0     via 10.2.2.2        cost 1
                                                  via 20.2.2.2        cost 2

                IP config>
```

route-policy *route-policy-identifier use-strictly-linear-policy*

Adds a route filter policy. A route filter policy consists of entries that define a set of routes that can be filtered to be included or excluded from the routing table of an external routing protocol such as OSPF or RIP.

route-policy-identifier

A string that identifies a route filter policy.

Valid Values: any 1-to-15-character ASCII string

Default Value: none

use-strictly-linear-policy

Yes indicates that matching will be done based strictly upon the sequence of index numbers of the route filter policy entries. The entry with the lowest index number will be processed first. *No* indicates that matching will be done using the longest-match application. The entry with the lower index number will be chosen only when more than one entry has the same address and mask.

Valid Values: Yes or No

Default Value: No

route-table-filter *destination mask [both | exact | more-specific] [exclusive | inclusive]*

Adds a route table filter for the specified routes. When **route-table-filtering** is enabled, the route-table-filter will be matched against routes added to the IP route table. The order in which route-table-filters is unimportant. Rather, the route-table-filter with the most specific match is chosen. If no match is found, the route is added to the route table. When **exact** is specified, the route destination and mask must be exactly the same as the route-table-filter destination and mask for a match to occur. When **more-specific** is specified, the route destination and mask must part of the range subsumed by the route-table-filter destination and mask. Specifying **both** is the superset of both and more-specific (that is, a match will occur in both the case of an exact match and a more-specific match). If the route-table-filter indicates **include**, the route will be added to the IP route table. If the route-table-filter indicates **exclude**, the route will not be added to the IP route table. Static and direct routes are never excluded from the IP route table.

destination mask

Valid Values: any valid IP mask

Default Value: both exclude

vrid ... Adds a Virtual Router ID definition for a VRRP router on a LAN segment.

interface-ip-address

Indicates the IP interface for which this VRID is being defined.

Valid Values: Any configured IP interface.

Default Value: none

vrid The Virtual Router identifier. The combination of the

ip-interface-address and *vrid* uniquely define the VRID. The same *vrid* can be used on more than one physical interface. If the VRID already exists, it will be modified.

Valid Values: 1-255

Default Value: none

advertisement-interval
The interval between VRRP advertisements.

Valid Values: 1-255

Default Value: 1

backup-router
Indicates whether this router is the master or a backup router for this VRID.

Valid Values: Yes or No

Default Value: No

backup-ip-address
Indicates the first IP address that is the backup for this VRID. Additional addresses may be added using the *add vr-address* command for LAN segments supporting more than one subnet. It is not applicable if **No** was configured for *backup-router*.

Valid Values: Any valid IP address.

Default Value: none

priority
Indicate the VRRP priority for backup routers. If a backup router takes over for the primary router, it will use this priority in its VRRP advertisements. It is not applicable if **No** was configured for *backup-router*. A master router will always advertise a priority of 255.

Valid Values: 1-254

Default Value: 100

functional/group mode
Indicates whether or not a multicast MAC address is used as the VRID virtual MAC address. All routers configured for this VRID should have the same value for this parameter in order for VRRP to function correctly. This parameter defaults to **No** and is not displayed if *hardware MAC mode* is configured as **Yes**.

Valid Values: Yes or No

Default Value: No

authentication-type
Indicates the type of authentication used for VRRP advertisements. The choices for authentication types are 1, which indicates a simple password; or 0, which indicates that no authentication is used.

Valid Values: none, simple

Default Value: none

authentication-key
The parameter that defines the password for this VRID. When

password authentication is used, only packets with the correct authentication key are accepted. The *authentication key* is not applicable when *none* is specified or defaulted for *authentication type*.

Valid Values: Any 1 - 8 characters.

Default Value: A null string.

vr-address ...

Adds a secondary address to a configured Virtual Router ID (VRID) definition. Secondary addresses will be included in VRRP advertisements for the VRID. Secondary addresses are necessary on physical LANs supporting multiple IP subnets. Each address designates the default gateway address for that subnet. If the router is a master router, addresses added using the *add vr-address* command will be advertised in addition to the *ip-interface-address* for the VRID. If the router is a backup router for the VRID, addresses added using the *add vr-address* command will be advertised in addition to the *backup-ip-address*.

interface-ip-address

The IP interface for the VRID.

Valid Values: Any configured IP interface.

Default Value: none

vrid The Virtual Router identifier. The combination of the *ip-interface-address* and *vrid* uniquely define the VRID. The VRID must be configured for addresses to be added to its definition. A master router and its backup routers must both be configured with the same VRID.

Valid Values: 1 to 255

Default Value: none

ip-address

The additional IP address that will be included in VRRP advertisements for the VRID.

Valid Values: Any IP address.

Default Value: none

Example: `add vr-address`

```
IP config>add vr-address
IP Interface [ ]? 153.2.2.25
Virtual Router ID (1-255) [0]? 1
Additional IP Address [ ]? 5.1.1.1
VRID 153.2.2.25/1 address 5.1.1.1 added successfully.
```

Change

Use the **change** command to change an IP configuration item previously installed by the **add** command. In general, you must specify the item you want to change, just as you specified the item with the **add** command.

Syntax:

change access-control . . .

address . . .

route . . .

access-control *record-number type IP-source source-mask IP-dest dest-mask* **[***first-protocol last-protocol***] [***first-port last-port***]**

Modifies an existing global access-control record. Use the **list access-control** command to view all existing records and obtain the record number.

Example: `change access-control 2`

```
Enter type [E]? i
Internet source [1.1.1.1]?
Source mask [255.255.255.255]?
Internet destination [2.2.2.2]?
Destination mask [255.255.255.255]?
Enter starting protocol number [6]?
Enter ending protocol number [6]?
Enter starting port number [23]?
Enter ending port number [23]?
```

address *old-address new-address new-mask*

Modifies one of the router's IP interface addresses. You must specify each new address together with the new address' subnet mask. This command can also be used to change an existing address' subnet mask.

Valid IP addresses:

- The class A range is 1.0.0.1 through 126.255.255.254
- The class B range is 128.0.0.1 through 191.255.255.254
- The class C range is 192.0.0.1 through 223.255.255.254

old-address

Valid Value: a currently configured IP interface address

Default Value: none

new-address

Valid Value: any valid IP address

Default Value: none

new-mask

Valid Value: 0.0.0.0 - 255.255.255.255

Default Value: none

Example: `change address 192.9.1.1 128.185.123.22 255.255.255.0`

route *dest-addr dest-mask new-next-hop1 new-cost1 [new-next-hop2 new-cost2 [new-next-hop3 new-cost3 [new-next-hop4 new-cost4]]]*

Modifies either the next hops or the costs associated with the configured static routes to the specified destination. The effect of this command is immediate; you do not have to reboot the router for it to take effect.

dest-addr

Valid Values: any valid IP address

Default Value: none

dest-mask

Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: none

new-next-hop

Valid Values: any valid IP address

Default Value: none

new-cost

Valid Values: an integer in the range 0 to 255

Default Value: 1

Example:

```
IP config>list routes

route to 1.1.0.0        ,255.255.0.0     via 10.1.1.1        cost 1
                                         via 20.1.1.1        cost 2
                                         via 30.1.1.1        cost 3
route to 2.2.0.0        ,255.255.0.0     via 10.2.2.2        cost 1
                                         via 20.2.2.2        cost 2

IP config>change route
IP destination []? 1.1.0.0
Address mask [255.0.0.0]? 255.255.0.0
Via gateway 1 at [.10.1.1.1]? 10.10.10.1
Cost [1]? 10
Via gateway 2 at [20.1.1.1]? 20.20.20.1
Cost [2]? 20
Via gateway 3 at [30.1.1.1]? 30.30.30.1
Cost [3]? 30
Via gateway 4 at []? 40.40.40.1
Cost [1]? 40
IP config>change route 2.2.0.0 255.255.0.0 10.10.10.2 10
IP config>list routes

route to 1.1.0.0        ,255.255.0.0     via 10.10.10.1      cost 10
                                         via 20.20.20.1      cost 20
                                         via 30.30.30.1      cost 30
                                         via 40.40.40.1      cost 40
route to 2.2.0.0        ,255.255.0.0     via 10.10.10.2      cost 10
```

Delete

Use the **delete** command to delete an IP configuration item previously installed by the **add** command. In general, you must specify the item you want to delete, just as you specified the item with the **add** command.

Syntax:

delete
- accept-rip-route . . .
- access-control . . .
- address . . .
- bootp-server
- default network/subnet-gateway . . .
- distributed default gateway
- filter . . .
- packet-filter
- redundant default gateway
- route . . .
- route-table-filter
- vrid . . .
- vr-address . . .

accept-rip-route *net-number*

Removes a route from the list of networks that the RIP protocol always accepts.

Valid Values: Any IP address contained in the list of accepted networks.

Default Value: none

Example: `delete accept-rip-route 10.0.0.0`

access-control *rule-number*

Deletes one of the access control rules from the global access control list.

Example: `delete access-control 2`

address *ip-interface-address*

Deletes one of the router's IP interface addresses.

Valid Values: any valid IP address

Default Value: none

Example: `delete address 128.185.123.22`

bootp-server *server-IP-address*

Removes a BOOTP server from an IP configuration.

Valid Values: any configured BOOTP server IP address

Default Value: 0.0.0.0

Example: `delete bootp-server 128.185.123.22`

default network/subnet-gateway *[ip-network-address]*

Deletes either the default gateway or the default subnet gateway for the specified subnetted network.

Valid Values: any valid IP address

Default Value: 0.0.0.0

Example: `delete default subnet-gateway 128.185.0.0`

distributed *interface-number*

Deletes the distributed IP Gateway from a LEC interface.

interface-number

Valid Values: Interface numbers of LECs with a distributed IP Gateway.

Default Value: none

Example:

```
Enter the Net number of distributed Gateway to delete:? 1
Gateway deleted.
```

filter *dest-addr dest-mask*

Deletes one of the router's filtered networks. The effect of this command is immediate; you do not have to reboot the router for it to take effect.

dest-addr

Valid Values: any valid IP address

Default Value: 0.0.0.0

dest-mask

Valid Values: 0.0.0.0 - 255.255.255.255

Default Value: none

Example: `delete filter 127.0.0.0`

```
Address mask [0.0.0.0]? 255.0.0.0
```

packet-filter *filter-name*

Deletes a specified packet-filter from the router's configuration.

Valid Values: any 16-character name.

You can include dashes (-) and underscores (_) in the name.

Default Value: none

Example:

```
IP config> delete packet-filter pf-in-0
All access controls defined for 'pf-in-0' will also be deleted.
Are you sure you want to delete (Yes or [No]): y
Deleted
IP config>
```

redundant *interface-number*

Deletes the Redundant IP Gateway from a LEC interface.

interface-number

Valid Values: Interface numbers of LECs with a Redundant Default IP Gateway.

Default Value: none

Example:

```
Enter the Net number of Redundant Gateway to delete:? 1
Gateway deleted.
```

route *dest-addr dest-mask [delete-next-hop1 [delete-next-hop2 [delete-next-hop3 [delete-next-hop4]]]]*

Deletes one of the device's configured static routes. The effect of this command is immediate; you do not have to reboot the router for it to take effect.

dest-addr

Valid Values: any valid IP address

Default Value: none

dest-mask

Valid Values: any valid IP mask

Default Value: none

delete-next-hop

Valid Values: Yes or No

Default Value: No

Example:

```
IP config>list routes

route to 1.1.0.0        ,255.255.0.0     via 10.10.10.1      cost 10
                                         via 20.20.20.1      cost 20
                                         via 30.30.30.1      cost 30
                                         via 40.40.40.1      cost 40
route to 2.2.0.0        ,255.255.0.0     via 10.10.10.1      cost 10

IP config>delete route 1.1.0.0 255.255.0.0
Delete gateway 10.10.10.1? [No]:
Delete gateway 20.20.20.1? [No]: y
Delete gateway 30.30.30.1? [No]:
Delete gateway 40.40.40.1? [No]: y
IP config>delete route 2.2.0.0 255.255.0.0
IP config>delete route 1.1.0.0 255.255.0.0 n y
IP config>list routes

route to 1.1.0.0        ,255.255.0.0     via 10.10.10.1      cost 10

IP config>
```

route-table-filter *destination mask mask-definition[both | exact | more specific]*

Deletes a route filter from the route table filters added using **add route-table-filter.** See “route-table-filter” on page 316 for the command extension definitions.

destination

Valid Values: any valid IP mask

Default Value: none

mask **Valid Values**: any valid IP mask

Default Value: none

mask-definition

Valid Values: any valid IP mask

Default Value: none

Example: `delete route-table-filter`

```
IP config>delete route-table-filter
Route Filter IP address []? 7.0.0.0
Route Filter IP mask []? 255.0.0.0
Enter Match type (B, E, or M) [B]?
Enter Definition type (I or E) [E]?
Route filter deleted
IP config>
```

vrid *interface-ip-address vrid*

Deletes a configured Virtual Router ID definition for a VRRP router.

interface-ip-address

Indicates the IP interface for which this VRID is being deleted.

Valid Values: Any configured IP interface.

Default Value: none

vrid The Virtual Router identifier. The combination of the *ip-interface-address* and *vrid* uniquely define the VRID. It is used to identify the VRID which is going to be deleted.

Valid Values: 1-255

Default Value: none

Example:

```
IP config>delete vrid
IP Interface [ ]? 153.2.2.25
Virtual Router ID (1-255) [0]? 1
VRID 153.2.2.25/1 deleted.
```

vr-address *interface-ip-address vrid ip-address*

Deletes a secondary address from a configured Virtual Router ID (VRID) definition.

interface-ip-address

The IP interface for the VRID.

Valid Values: Any configured IP interface.

Default Value: none

vrid The Virtual Router identifier. The combination of the *ip-interface-address* and *vrid* uniquely define the VRID. The VRID must be configured for addresses to be deleted from its definition.

Valid Values: 1-255

Default Value: none

ip-address

The additional IP address that will be deleted from the VRRP definition.

Valid Values: Any IP address.

Default Value: none

Example:

```
IP config>delete vr-address
IP Interface [ ]? 153.2.2.25
Virtual Router ID (1-255) [0]? 1
IP Address to delete [ ]? 5.1.1.1
VRID 153.2.2.25/1 addr 5.1.1.1 deleted.
```

Disable

Use the **disable** command to disable IP features previously enabled by the **enable** command.

Syntax:

disable

- arp-net-routing
- arp-subnet-routing
- bootp-forwarding
- classless
- directed-broadcast
- echo-reply
- fragment-offset-check
- icmp-redirect . . .
- nexthop-awareness . . .
- override default/static-routes . . .
- packet-filter
- receiving rip . . .
- receiving dynamic all/hosts/nets/subnets . . .
- record-route
- rip
- rip2
- route-table-filtering
- same-subnet
- sending all/default/net/subnet/poisoned/host/static/...
- sending rip1-routes-only
- source-routing
- tftp-server
- timestamp
- udp-forwarding . . .

vrrp . . .

arp-net-routing
Turns off ARP network routing. When this is enabled, the router replies by proxy to all ARP requests for remote destinations that are best reached through the router. This is the default and the generally recommended setting.

Example: `disable arp-net-routing`

arp-subnet-routing
Turns off the IP feature called ARP subnet routing or proxy ARP, which, when enabled, deals with hosts that have no IP subnetting support. This is the default and the generally recommended setting.

Example: `disable arp-subnet-routing`

bootp-forwarding
Turns off the BOOTP/DHCP relay function.

Example: `disable bootp-forwarding`

classless
Disables the suppression of natural network routes. Natural network routes (for example, class A, B, or C routes) will be automatically generated for advertisement in protocols that do not advertise the subnet mask (for example, RIPv1).

directed-broadcast
Disables the forwarding of IP packets whose destination is a non-local (for example, remote LAN) broadcast address. The source host originates the packet as a unicast where it is then forwarded as a unicast to a destination subnet and “exploded” into a broadcast. You can use these packets to locate network servers.

Note: Forwarding and exploding cannot be disabled separately.

Example: `disable directed-broadcast`

echo-reply
Disables the router’s ICMP Echo Reply function. Thus a ping sent to any of the router’s interfaces will not generate a reply. The router defaults to echo-reply enabled.

Example: `disable echo-reply`

fragment-offset-check
Disables the checking of the fragment offset of received IP packets. When this check is enabled, the router checks each fragment to ensure that no secondary fragment has overlaid the first eight bytes of the first fragment’s payload. By default this check is disabled.

icmp-redirect *ip-interface-address*
Disables the router from sending ICMP Redirect messages on the specified IP interface. If you enter nothing at the prompt for the IP interface address, the router will be disabled from sending ICMP Redirect messages on all IP interfaces.

ip-interface-address
Valid Values: any valid IP address

Default Value: none

Example:

```
IP config> disable icmp-redirect
Interface address (NULL for all) []? 192.9.200.44
IP config>
```

nexthop-awareness *ip-interface-address*

Disables nexthop awareness on an IP interface.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example:

```
IP config>disable nexthop-awareness 1.1.1.1
IP config>disable nexthop-awareness
Interface address []? 2.2.2.2
IP config>
```

override default/static-routes *ip-interface-address*

By default, routes received by RIP do not override static routes. However, the command **enable override static-routes** enables routes received by RIP to override static routes. After RIP routes have been enabled to override static routes, you can use the command **disable override default-route** or **disable override static-route** to again prevent static routes from being overridden by routes received by RIP. The command **disable override default-route** prevents a default route received by RIP on interface *ip-interface-address* from replacing a default route already installed in the IP routing table. The command **disable override static-routes** prevents RIP routes received on interface *ip-interface-address* from overriding any of the router's static routes.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `disable override default 128.185.123.22`

packet-filter *filter-name*

Disables specified interface-specific access control list (packet-filters).

filter-name

Valid Values: Any 16-character name. You can include dashes (-) and underscores (_) in the name.

Default Value: None

Example: `disable packet-filter pf-in-0`

receiving rip *ip-interface-address*

Prevents RIP from processing any RIP updates received on interface *ip-interface-address.*

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `disable receiving rip 128.185.123.22`

receiving dynamic all/hosts/nets/subnets *ip-interface-address*

The **disable receiving dynamic nets** command ensures that for RIP updates received on the interface *ip-interface-address*, the router accept

only those network level routes entered by the **add accept-rip-route** command. The **disable receiving dynamic subnets** command produces the analogous behavior for subnet routes. The **disable receiving dynamic host** produces the analogous behavior for host routes.

ip-interface-address
Valid Values: any valid IP address

Default Value: none

Example: `disable receiving dynamic nets 128.185.123.22`

record-route
Disables the router from receiving or forwarding IP packets that contain a record route IP option. By default, the router receives and forwards these packets.

rip Turns off the RIP protocol.

Example: `disable rip`

rip2 Disables RIP2 on an IP interface on which it was previously enabled.

ip-interface-address
Indicates the IP interface on which RIP2 is disabled.

Valid Values: any valid IP address

Default Value: none

Example: `disable rip2 128.185.123.22`

route-table-filtering
Disables application of route-table-filters when routes are added to the routing table.

Example: `disable route-table-filtering`

same-subnet
Disables the same subnet option. When the router is rebooted, it will not allow multiple IP interfaces to the same subnet to be installed. This is the default.

Example: `disable same-subnet`

sending all/default/host/net/poisoned/static/subnet *ip-interface-address*
Prevents the router from advertising the specified type of route in RIP updates sent out using the interface ip-interface-address. The other flags that control the RIP routes sent out an interface are **host-routes**, **static-routes**, **net-routes**, and **subnet-routes**. You can turn these off individually. A route is advertised if it is specified by any of the enabled flags.

ip-interface-address
Valid Values: any valid IP address

Default Value: none

Example: `disable sending net-routes 128.185.123.22`

sending rip-routes-only *ip-interface-address*
Stops advertising only RIP routes in the RIP2 multicast packets.

ip-interface-address

Valid Values: any valid IP address of an interface that has RIP2 enabled.

Default Value: none

Example: `disable sending rip1-routes-only 128.185.123.22`

source-routing

Prevents the router from forwarding source-routed packets (that is, IP packets that include a source-route option). This option defaults to source-routing enabled.

Example: `disable source-routing`

tftp-server

Prevents the router from accepting TFTP GET or PUT requests from the network. This prevents the inadvertent overlaying of configuration files or load images from another device. You will still be able to perform TFTP client operations (GETs and PUTs) from the router through a directly attached terminal or telnet session.

timestamp

Disables the router from receiving or forwarding IP packets that contain a timestamp IP option. By default, the router receives and forwards these packets.

udp-forwarding *port-number*

Disables UDP forwarding for packets received by the router with the specified UDP destination port number.

Default: UDP forwarding is disabled for all port numbers.

port-number

Valid Values: an integer in the range 0 to 65535

Default Value: 0

Example: `disable udp-forwarding 36`

vrrp Disables Virtual Router Redundancy Protocol.

Example: `disable vrrp`

Enable

Use the **enable** command to activate IP features, capabilities, and information added to your IP configuration.

Syntax:

enable arp-net-routing
arp-subnet-routing
bootp-forwarding
classless
directed-broadcast
echo-reply
icmp-redirect

nexthop-awareness

override default ...

override static-routes ...

packet-filter

receiving rip ...

receiving dynamic all ...

receiving dynamic hosts...

receiving dynamic nets ...

receiving dynamic subnets ...

record-route

rip

rip2

route-table-filtering

same-subnet

sending all-routes ...

sending default-routes ...

sending host-routes ...

sending net-routes ...

sending poisoned-reverse-routes

sending rip1-routes-only

sending static-routes ...

sending subnet-routes ...

source-routing

tftp-server

timestamp

udp-forwarding ...

vrrp ...

arp-net-routing

Turns on ARP network routing. When enabled, the router replies by proxy to all ARP requests for remote destinations that are best reached through the router. Use this command when there are hosts on the LAN that ARP for all destinations, instead of (as is proper) only local destinations.

Example: `enable arp-net-routing`

arp-subnet-routing

Turns on the router's ARP subnet routing (sometimes also called Proxy ARP) function. This function is used when there are hosts unaware of subnetting attached to directly connected IP subnets. The directly connected subnet having subnet-incapable hosts must use ARP for this feature to be useful.

The way ARP subnet routing works is as follows. When a subnet-incapable host wants to send an IP packet to a destination on a remote subnet, it

does not realize that it should send the packet to a router. The subnet-incapable host therefore simply broadcasts an ARP request. This ARP request is received by the router. The router responds as the destination (hence the name proxy) if both arp-subnet-routing is enabled and if the next hop to the destination is over a different interface than the interface receiving the ARP request.

If there are no hosts on your LAN that are “subnet-incapable,” do not enable ARP-subnet routing. If ARP subnet routing is needed on a LAN, it should be enabled on all routers on that LAN.

Example: `enable arp-subnet-routing`

bootp-forwarding

Turns on BOOTP/DHCP packet forwarding. In order to use BOOTP forwarding, you must also add one or more BOOTP servers with the **add bootp-server** command.

Example: `enable bootp-forwarding`

```
Maximum number of forwarding hops [4]?
Minimum seconds before forwarding [0]?
```

Maximum number of forwarding hops

Maximum number of allowable BOOTP agents that can forward a BOOTP request from the client to the Server (this is not the maximum number of IP hops to the server).

Default: 4

Minimum seconds before forwarding

This parameter is generally not used. Use this parameter when there is a redundant path between the client and the server, and you want to use the secondary path or paths as a standby.

Default Value: 0

classless

Indicates the router will be operating in a classless IP addressing environment. The IBM 8371 fully supports CIDR addressing as described in RFC 1817 without this option enabled. Enabling this option prevents automatic generation of the natural network routes (for example, Class A, B, or C network routes) corresponding to routes added to the IP route table. If you are not running RIPv1 you do not require the natural network route.

Example: `enable classless`

directed-broadcast

Enables the forwarding of IP packets whose destination is a network-directed or subnet-directed broadcast address. The packet is originated by the source host as a unicast where it is then forwarded as a unicast to a destination subnet and “exploded” into a broadcast. These packets can be used to locate network servers. This command enables both the forwarding and exploding of directed broadcasts. The IP packet forwarder never forwards link level broadcasts/multicasts, unless they correspond to Class D IP addresses. (See the OSPF **enable multicast-routing** command.) The default setting for this feature is enabled.

Note: Forwarding and exploding cannot be implemented separately. Also, the router will not forward all-subnets IP broadcasts.

Example: `enable directed-broadcast`

echo-reply

Enables the building and sending of an ICMP Echo Reply in response to an ICMP Echo Request.

Example: `enable echo-reply`

icmp-redirect *ip-interface-address*

Enables the router to send ICMP Redirect messages on the specified IP interface. If you enter nothing at the prompt for the IP interface address, the device will be enabled to send ICMP Redirect messages on all IP interfaces.

ip-interface-address

Valid Values: any valid IP address, or nothing for all IP interfaces

Default Value: none

Example:

```
IP config> enable icmp-redirect
Interface address (NULL for all) []? 192.9.200.44
IP config>
```

nexthop-awareness *ip-interface-address*

Enables nexthop awareness on an IP interface.

ip-interface-address

Valid Values: any valid IP address

Default Value: disabled

Example:

```
IP config>enable nexthop-awareness 1.1.1.1
IP config>enable nexthop-awareness
Interface address []? 2.2.2.2
IP config>
```

override default *ip-interface-address*

Enables received RIP information to override any default route installed in the IP routing table. This command is invoked on a per-IP-interface basis. When the **enable override default** command is invoked, default RIP routes received on interface ip-interface-address overwrites the router's current default route, providing the cost of the new default is cheaper.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable override default 128.185.123.22`

override static-routes *ip-interface-address*

Enables received RIP information to override some of the router's statically configured routing information. This command is invoked on a per-IP-interface basis. When the **enable override static-routes** command is invoked, RIP routing information received on interface ip-interface-address overwrite statically configured network/subnet routes providing the cost of the RIP information is cheaper.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable override static-routes 128.185.123.22`

packet-filter *filter-name*

Enables specified interface-specific access control list (packet-filters).

filter-name

Valid Values: any 16-character name. You can include dashes (-) and underscores (_) in the name.

Default Value: none

Example: `enable packet-filter pf-in-0`

receiving rip *ip-interface-address*

Enables the processing of RIP updates that are received on a particular interface. This command has an analogous disable command. (See the **disable receiving** command.) This command is enabled by default.

If you invoke the **disable receiving rip** command, no RIP updates will be accepted on interface *ip-interface-address* address.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable receiving rip 128.185.123.22`

receiving dynamic nets *ip-interface-address*

Modifies the processing of RIP updates that are received on a particular interface. This command has an analogous disable command. (See the **disable receiving** command.) This command is enabled by default.

If you invoke the **disable receiving dynamic nets** command, for RIP updates received on interface *ip-interface-address*, the router will not accept any network-level routes unless they have been specified in an **add accept-rip-route** command.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable receiving dynamic nets 128.185.123.22`

receiving dynamic subnets *ip-interface-address*

Modifies the processing of RIP updates that are received on a particular interface. This command has an analogous disable command. (See the **disable receiving** command.) This command is enabled by default.

If you invoke the **disable receiving dynamic subnets** command, for RIP updates received on interface *ip-interface-address*, the router will not accept any subnet-level routes unless they have been specified in an **add accept-rip-route** command.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable receiving dynamic subnets 128.185.123.22`

record-route

Enables the router to receive and forward IP packets that contain a record route IP option. This is the default.

Note: After it has been enabled, this function can be activated without affecting any other functions of IP. See the talk 5 **reset IP** command for more information.

rip Enables the router's RIP protocol processing.

When RIP is enabled, the following default behavior is established:

- The router includes all network and subnet routes in RIP updates sent out on each of its configured IP interfaces.
- The router processes all RIP updates received on each of its configured IP interfaces.

To change any of the default sending/receiving behaviors, use the IP configuration commands, which are defined on a per-IP-interface basis.

Example: `enable rip`

rip2 *ip-interface-address RIP2-authentication authentication-keys*

Enables RIP2 on an IP interface. RIP2 advertisements are sent to the 224.0.0.9 multicast address. RIP2 is described in RFC 1723.

ip-interface-address

Indicates the IP interface on which RIP2 is enabled.**Valid Values**: any valid IP address

Default Value: none

RIP2-authentication

Indicates whether or not a simple clear-text key will be used for RIP2 authentication. Authentication is not required.**Valid Values**: yes or no

Default Value: yes

authentication-key

Defines a clear-text password which will be used for RIP2 authentication. You are prompted for this string only when you answer **yes** to the question "Set RIP-2 Authentication?" When RIP2 authentication is used, only RIP2 packets with a matching password are accepted.**Valid Values**: a clear-text ASCII string

Default Value: a null string

Example:

```
IP config>enable rip2
Set for which interface address [0.0.0.0]? 153.2.2.25
RIP2 is enabled on this interface.
Set RIP-2 Authentication? [Yes]: yes
Authentication Key []? C1C3C5C5
Retype Auth. Key []? C1C3C5C5
RIP2 Authentication is enabled on this interface.
```

route-table-filtering

Applies route table filters to any route added to the routing table. Route table filters are applied based on a most-specific match of the destination and network mask. Route table filters are never applied to direct routes or static routes.

Example: `enable route-table-filtering`

same-subnet

Enables the same subnet option. When the device is rebooted, it will allow

multiple IP interfaces to the same subnet to be installed. Multiple IP interfaces to the same subnet are useful under only one of the following conditions:

- OSPF Point-to-Multipoint is configured on the IP interfaces.
- Nexthop Awareness is enabled on the IP interfaces, and static routes are defined for the routes that go through the IP interfaces.

By default, this option is disabled.

Example: `enable same-subnet`

sending default-routes *ip-interface-address*

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous disable command. (See the **disable sending** command.) The effect of the **enable sending** command is additive. Each separate enable sending command specifies that a certain set of routes should be advertised from a particular interface. A route is included in a RIP update only if it has been included by at least one of the enable sending commands. The **enable sending default-routes** command specifies that the default route (if one exists) should be included in RIP updates sent out interface ip-interface-address.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable sending default-routes 128.185.123.22`

Note: By default, RIP will send network, subnet, and static routes.

sending net-routes *ip-interface-address*

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous disable command. (See the **disable sending** command.)

The effect of the **enable sending** command is additive. Each separate **enable sending** command specifies that a certain set of routes should be advertised from a particular interface. A route is included in an RIP update only if it has been included by at least one of the **enable sending** commands. The **enable sending network-routes** command specifies that all network-level routes should be included in RIP updates sent out interface *ip-interface-address*. A network-level route is a route to a single class A, B, or C IP network.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable sending net-routes 128.185.123.22`

sending poisoned-reverse-routes *ip-interface-address*

A technique used by RIP to improve convergence time when routes change (for complete details on the technique, refer to RFC 1058). Use of this technique increases the size of RIP update messages. You may find it more acceptable to minimize routing overhead by accepting somewhat slower convergence. The **disable sending poisoned-reverse-routes** command

specifies that poisoned reverse routes should not be included in RIP updates sent out on an interface specified by the **enable ip-interface-address** command.

Default: Enabled

ip-interface-address

Valid Values: any valid IP address

Default Value: none

sending rip-routes-only *ip-interface-address*

To advertise only RIP routes in the RIP2 multicast packets.

ip-interface-address

Valid Values: any valid IP address of an interface that has RIP2 enabled.

Default Value: none

Example: `enable sending rip-routes-only 128.185.123.22`

sending subnet-routes *ip-interface-address*

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous disable command. (See the **disable sending** command.) The effect of the **enable sending** command is additive. Each separate **enable sending** command specifies that a certain set of routes should be advertised out a particular interface. A route is included in an RIP update only if it has been included by at least one of the enable sending commands. The **enable sending subnet-routes** command specifies that all subnet routes should be included in RIP updates sent out interface ip-interface-address. However, a subnet route is included only if ip-interface-address connects directly to a subnet of the same IP subnetted network.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable sending subnet-routes 128.185.123.22`

sending static-routes *ip-interface-address*

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous disable command. (See the **disable sending** command.) The effect of the **enable sending** command is additive. Each separate **enable sending** command specifies that a certain set of routes that meet other sending criteria should be advertised out a particular interface. A route is included in an RIP update only if it has been included by at least one of the **enable sending** commands. The **enable sending static-routes** command specifies that all statically configured and directly connected routes should be included in RIP updates sent out interface *ip-interface-address*.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

Example: `enable sending static-routes 128.185.123.22`

sending host-routes *ip-interface-address*

Determines the contents of RIP updates that are sent out a particular interface. This command has an analogous **disable ...** command. (See the **disable sending** command.) The effect of the **enable sending** command is additive. Each separate **enable sending** command specifies that a certain set of routes should be advertised out a particular interface. A route is included in an RIP update only if it has been included by at least one of the **enable sending** commands. The **enable sending host-routes** command specifies that all host routes should be included in RIP updates sent out interface ip-interface-address.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

source-routing

Allows the router to forward IP packets containing an IP source route option.

Example: `enable source-routing`

tftp-server

Allows the router to accept TFTP GET or PUT requests from the network for configuration files or image loads.

Example: `enable tftp-server`

timestamp

Enables the router to receive and forward IP packets that contain a Timestamp IP option. This is the default.

Note: After it has been enabled, this function can be activated without affecting any other functions of IP. See the talk 5 **reset IP** command for more information.

udp-forwarding *port-number*

Enables UDP forwarding for packets received by the router with the specified UDP destination port number.

Default: UDP forwarding is disabled for all port numbers.

port-number

Valid Values: an integer in the range 0 to 65535

Default Value: 0

Example: `enable udp-forwarding 36`

vrrp Enables Virtual Router Redundancy Protocol

Example: `enable vrrp`

List

Use the **list** command to display various pieces of the IP configuration data, depending on the particular subcommand invoked.

Syntax:

list all

access-control

addresses

bootp

distributed default gateway

filters

icmp-redirect

mtu

nexthop-awareness

packet-filter

parameters

protocols

redundant default gateway

rip

route-table-filtering

routes

sizes

tags

udp-forwarding

vrid

all Displays the entire IP configuration.

Example: `list all`

access-control
Displays the configured access control mode (enabled or disabled) and the list of configured global access control records. Each record is listed with its record number. This record number can be used to reorder the list with the IP **move access-control** command.

Example: `list access-control`

```
list access-control
1   Type=I   Source=0.0.0.0    Dest   =0.0.0.0     Prot=17
             SMask =0.0.0.0    DMAask =0.0.0.0
             SPorts=5004-5511  DPorts=5004-5511
                    T/C=**/**  Log=N
             BypassComp BypassEnc
```

addresses
Displays the IP interface addresses that have been assigned to the router, along with their configured broadcast formats. The interface identified by *BDG/0* is the bridging interface.

Example: `list addresses`

bootp Indicates whether BOOTP forwarding is enabled or disabled as well as the configured list of BOOTP servers.

Example: `list bootp`

distributed default gateway
Displays the distributed IP Gateway for each interface configured.

Example: `list distributed`

```
Distributed IP Gateways for each interface:
   inf 4  11.1.1.6     255.0.0.0     00.00.00.00.00.BA  primary
   inf 8  33.3.3.6     255.0.0.0     00.00.00.00.00.AB  backup
```

filters Lists the router's configured filtered networks.

icmp-redirect
Lists whether the sending of ICMP redirect messages is enabled or disabled on each IP interface.

mtu Lists configured MTU values.

nexthop-awareness
Lists the setting of nexthop awareness on all IP interfaces.

Example:

```
IP config>list nexthop-awareness
Nexthop awareness for each IP interface address:
   intf  0   1.1.1.1          255.0.0.0        nexthop awareness enabled
   intf  1   2.2.2.2          255.0.0.0        nexthop awareness disabled
IP config>
```

packet-filter *filter-name*
Lists information on packet filters. If you specify a name, the command lists access control information configured for the filter. If you do not specify a filter name, the command lists configured packet-filters.

Example: `list packet-filter pf-in-0`

```
Name              Direction     Interface
pf-in-0           In            0

Access Control is: enabled

List of access control records:



2  Type=INS  Source=10.1.1.1      Dest=10.1.1.2          Prot=0-255
             Mask=255.255.255.255 Mask=255.255.255.254
             Sports=  N/A         Dports=  N/A           Tid=5279
                                  Log=Yes ELS=N SNMP=Y SLOG=L(Emergency)

3  Type=I    Source=0.0.0.0       Dest=0.0.0.0          Prot=0-255
             Mask=0.0.0.0         Mask=0.0.0.0
             Sports= 1-65535      Dports=  1-68835
                                  Log=No
```

parameters
Lists the various global IP parameters.

Example: `list parameters`

```
IP config>list parameters
ARP-SUBNET-ROUTING    : enabled
ARP-NET-ROUTING       : enabled
CLASSLESS             : disabled
DIRECTED-BROADCAST    : enabled
ECHO-REPLY            : enabled
FRAGMENT-OFFSET-CHECK : enabled
REASSEMBLY-SIZE       : 12000 bytes
RECORD-ROUTE          : enabled
ROUTING TABLE-SIZE    : 768 entries (52224 bytes)
(Routing) CACHE-SIZE  : 64 entries
SAME-SUBNET           : disabled
SOURCE-ROUTING        : enabled
TIMESTAMP             : enabled
TTL                   : 64
```

protocols
Displays the configured state of the IP routing protocols (OSPF, RIP, BGP) along with other general configuration settings.

Example: `list protocols`

redundant default gateway
Displays the Redundant Default IP Gateway for each interface configured.

Example: `list redundant`

```
Redundant Default IP Gateways for each interface:
   inf 4  11.1.1.6     255.0.0.0     00.00.00.00.00.BA  primary
   inf 8  33.3.3.6     255.0.0.0     00.00.00.00.00.AB  backup
```

rip Displays all RIP configuration parameters. RIP can be configured to receive and send dynamic routes or the routes can be defined by a route filter policy. See the IP configuration commands **enable receiving dynamic nets/subnets/hosts** for more information about dynamic routing. See "Route Filter Policy Configuration" on page 349 for more information about route filter policies.

Example:

```
IP config>list rip

RIP: enabled
RIP default origination: disabled
RIP global receive policy: rip-in

  Per-interface address flags:
  Net:     0  153.2.2.25       RIP Version 1
                               Send net, subnet and static routes
                               Receive routes based on global receive
                                    policy: rip-in
                               RIP interface input metric: 1
                               RIP interface output metric: 0
  Net:     1  153.2.1.1        RIP Version 1
                               Send net, subnet and static routes
                               Receive routes based on global receive
                                    policy: rip-in
                               RIP interface input metric: 1
                               RIP interface output metric: 0
  Net:     2  0.0.0.2          RIP Version 1
                               Send routes based on interface send
                                    policy: rip-import
                               Receive routes based on global receive
                                    policy: rip-in
                               RIP interface input metric: 1
                               RIP interface output metric: 0

Accept RIP updates always for:
[NONE]
```

route-table-filtering

Displays the list of route filters added to the routing filter.

Example: `list route-table-filtering`

```
IP config>list route-table-filtering

Route Filtering Disabled

Destination     Mask           Match Type
10.1.1.0        255.255.255.0   BOTH  E
50.50.0.0       255.255.0.0     BOTH  I
10.1.1.1        255.255.255.255 EXACT I
50.0.0.0        255.0.0.0       BOTH  E

MORE-Match more-specific routes   EXACT-Match route exactly
BOTH-Match exact and more-specific routes  E-Exclude  I-Include
IP config>
```

routes

Displays the list of static routes that have been configured.

Example: `list routes`

```
IP config>list routes

route to 1.1.0.0        ,255.255.0.0     via 10.1.1.1        cost 1
                                         via 20.1.1.1        cost 2
                                         via 30.1.1.1        cost 3
route to 2.2.0.0        ,255.255.0.0     via 10.2.2.2        cost 10
route to 3.3.0.0        ,255.255.0.0     via 10.3.3.3        cost 100
                                         via 20.3.3.3        cost 200
```

sizes Displays the routing table size, reassembly buffer size, and the route cache size.

Example: `list sizes`

tags Displays the per-interface tags that will be associated with received RIP information. These tags can be used to group routes together for later readvertisement via BGP where a tag will be treated as if it were a route's source autonomous system (AS). Tags are also propagated by the OSPF routing protocol.

Example: `list tags`

udp-forwarding
Displays all the configured information for the UDP Forwarding function, including all ports and all IP addresses.

Example: `list udp-forwarding`

vrid Displays the configured VRRP status, VRIDs, and VRID addresses. In this example, the *preempt-mode* parameter and the *hardware MAC address* option are both **yes** as shown by the Flags field that displays P and H.

Example:

```
IP config>list vrid

VRRP Enabled

                              --VRID Definitions--

IP address      VRID  Priority Interval Auth   Auth-key Flags   Address(es)
153.2.2.25         1       255        1 None   N/A      P,H
```

Move

Use the **move** command to change the order of records in the global access control list. This command places record number *from#* immediately after record number *to#*. After you move the records, they are immediately renumbered to reflect the new order.

The router applies the access control records in a list in the order that they were created. For each packet received on an interface, the router applies each access control record in order until it finds a match. The first record that matches the packet determines whether it will be discarded, or forwarded to its destination.

This makes the order of the access control records very important. If they are in the wrong order, certain packets may slip through, or be blocked, in a manner contrary to your intentions.

Let us say, for example, that access control record 1 enforces the rule: *all packets from network 10.0.0.0 shall be blocked on this interface.* Contrary to this, access control record 2 states: *Packets from subnet 10.5.5.0 in network 10.0.0.0, which are destined for address 1.2.3.4, shall be allowed to pass.* Assigned in this order, these records will block all traffic from 10.0.0.0, even though record 2 explicitly allows certain types of packets to pass.

In this example, record 1 makes record 2 moot. Record 1 guarantees that the router discards all packets from 10.0.0.0, despite the intent of record 2, which is that certain packets be forwarded. The key to fixing this type of problem is in the order of the access control records. This way, packets in subnet 10.5.5.0 and destined for address 1.2.3.4 will pass through the interface; the router discards all other packets from 10.0.0.0 as intended.

Syntax:

move access-control *from# to#*

Example: `move 5 2`

Set

Use the **set** command to set certain values, routes, and formats within your IP configuration.

Syntax:

set access-control...
broadcast-address...
cache-size
default network-gateway...
default subnet-gateway...
internal-ip-address
mtu
originate-rip-default
reassembly-size
rip-in-metric
rip-out-metric
router-id
routing table-size
tag . . .
ttl

access-control *on* **or** *off*

Allows you to configure the router to enable or disable IP access control. Setting access-control *on* enables the global access control list as well as the interface–specific lists. Setting it *off* disables all lists but does not delete them

Example: `set access-control on`

broadcast-address *ip-interface-address style fill-pattern*

Specifies the IP broadcast format that the router uses when broadcasting packets out on a particular interface. IP broadcasts are most commonly used by the router when sending RIP update packets.

The style parameter can take either the value local wire or the value network. Local-wire broadcast addresses are either all ones (255.255.255.255) or all zeros (0.0.0.0). Network style broadcasts begin with the network and subnet portion of the ip-interface-address.

You can set the fill-pattern parameter to either 1 or 0. This indicates whether the rest of the broadcast address (that is, other than the network and subnet portions, if any) should be set to all ones or all zeros.

When receiving the router recognizes all forms of the IP broadcast address.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

style **Valid Values**: *local-wire* or *network*

Default Value: local-wire

fill-pattern
Valid Values: *0* or *1*

Default Value: 1

The example below configures a broadcast address of 255.255.255.255. The second example produces a broadcast address of 192.9.1.0, assuming that the network 192.9.1.0 is not subnetted.

Example: `set broadcast-address 192.9.1.11 local-wire 1 set broadcast-address 192.9.1.11 network 0`

cache-size *entries*
Configures the maximum number of entries for the IP routing cache. This cache stores information about the specific IP addresses to which the router has recently forwarded packets. The cache reduces the processing time needed to forward multiple packets to the same destination.

In contrast with this cache, the IP routing *table* stores information about all accessible networks but does not contain specific IP destination addresses. Use the **set routing table-size** command to configure the size of the IP routing table.

Valid Values: 64 to 10000

Default Value: 64

Example: `set cache-size 64`

default network-gateway *next-hop cost*
Configures a route to the authoritative router (default gateway). You should assume that the router's default gateway has more complete routing information than the router itself.

The route is specified by the IP address of the next hop (next-hop) and the distance (cost) to the default gateway.

All packets having unknown destinations are forwarded to the authoritative router (default gateway).

nexthop
Valid Values: any valid IP address

Default Value: 0.0.0.0 with a gateway cost of 1.

cost **Valid Values**: an integer in the range 0 to 255

Default Value: 1

Example: `set default network-gateway 192.9.1.10 10`

default subnet-gateway *subnetted-network next-hop cost*
Configures a route to a subnetted network's authoritative router (default subnet gateway). You can configure a separate default subnet gateway for each subnetted network.

The IP address of the next hop (next-hop) and the distance (cost) to the default subnet gateway specify the route.

All packets destined for unknown subnets of a known subnetted network are forwarded to the subnetted network's authoritative router (default subnet gateway).

subnetted network

Valid Values: any valid IP address

Default Value: 0.0.0.0

next-hop

Valid Values: any valid IP address

Default Value: 0.0.0.0

cost

Valid Values: an integer in the range 0 to 255

Default Value: 1

Example: `set default subnet-gateway 128.185.0.0 128.185.123.22 6`

internal-ip-address *ip-address*

Configures an IP address that is independent of the state of any interface. The internal address is always considered active. The primary reason for defining an internal address is to provide an address for a TCP connection that will not become inactive when an interface becomes inactive. This address is used for data link switching (DLSw), allowing alternate paths to be used to avoid disrupting DLSw connections when an interface becomes inactive. Because the internal address remains active and because OSPF maintains active IP routes to this destination, IP routing can switch DLSw traffic onto the alternate path without bringing down the TCP connection or disrupting the SNA sessions that are running on top of DLSw.

The internal IP address also provides some value when unnumbered interfaces are used. It is the first choice as a source address for packets originated by this router and transmitted over an unnumbered interface. The stability of this address makes it easier to keep track of such packets. The chance for confusion is further reduced when the same IP address is used for both the router ID and the internal address. Therefore the router ID will default to the internal address.

When an internal address is defined, it will be advertised by OSPF as a host route into all areas directly attached to the router. It will also show up as a host route and will be advertised in RIP if allowed by the RIP sending configuration of the interface.

Valid Values: any valid IP address.

Default Value: none

Example: `set internal-ip-address 142.82.10.1`

mtu Sets the MTU value for the IP protocol on this interface.

Valid Values: 0, 68 - 65535

Default Value: Minimum of all non-zero MTUs on the network

originate-rip-default

Causes RIP to advertise this router as the default gateway. Use this command in the following environment:

- The IP routes in this router's routing table are determined by a number of protocols.
- RIP is one of those protocols.
- At most partial routing information is imported from the other protocols and advertised by RIP.

Traffic in the RIP network for destinations that are not known by RIP can follow the default path to this router. The more complete routing information in this node's route table can then be used to forward the traffic along an appropriate path towards its destination. You can configure the router to only originate the default when routes are known to this router that will not be advertised in the RIP network.

When you issue this command, you will be prompted to indicate whether the router should always originate a RIP default or to originate a RIP default only when the route from other protocols are available.

This default route will direct traffic bound for a non-RIP network to a boundary router. Originating a single default route means that the boundary router does not have to distribute the other network's routing information to the other nodes in its network.

from AS number

Valid Values: an integer in the range 0 to 65535

Default Value: none

to network number

Valid Values: any valid IP address

Default Value: none

default cost

Valid Values: an integer in the range 0 to 255

Default Value: 1

Example: `set originate-rip-default`

```
IP config> set originate rip-default
Always originate default route? [No]:?
Originate default if BGP routes available? [No] yes
  From AS number [6]?
  To network number [0.0.0.0]?
Originate default if OSPF routes available? [No]
Originate default cost [1]?
```

- Answering "Yes" to the "Always originate" question means a default route is always originated.
- Answering "Yes" to the "BGP" question originates a default whenever there are BGP routes in the routing table.
- Answering "Yes" to the "if OSPF routes available" question causes the RIP default to be advertised when OSPF routes are in the routing table.
- When the router does decide to originate a RIP default, it uses the "original default cost" number.
- When 0 is specified for the BGP route AS (Autonomous System) number, a route meeting the network criteria from any AS will cause a RIP default to be originated.
- When 0.0.0.0 is specified for the BGP network criteria, any BGP route meeting AS criteria will cause a RIP default to be originated.

reassembly-size *bytes*

Configures the size of the buffers that are used for the reassembly of fragmented IP packets.

Valid Values: 2048-65535

Default: 12000

Example: `set reassembly-size 12000`

rip-in-metric *ip-interface-address metric*

Allows the configuration of the metric to be added to RIP routes of an interface prior to installation in the routing table.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

metric **Valid Values**: an integer in the range 1 to 15

Default Value: 1

Example: `set rip-in-metric 128.185.120.209 1`

rip-out-metric *ip-interface-address metric*

Allows the configuration of the metric to be added to RIP routes advertised on an interface configured to advertise RIP or RIP2 routes.

ip-interface-address

Valid Values: any valid IP address

Default Value: none

metric **Valid Values**: an integer in the range 0 to 15

Default Value: 0

Example: `set rip-out-metric 128.185.120.209 0`

router-id *ip-address*

Sets the default IP address used by the router when sourcing various IP packets. This address is of particular importance in OSPF.

The router ID must match one of the configured IP interface addresses of the router or the configured internal IP address. If not, it is ignored. When ignored, or just not configured, the default IP address of the router (and its OSPF router ID) is set to the internal IP address (if configured) or to the first IP address in the router's configuration.

Valid Values: any valid IP address

Default Value: none

Example: `set router-id 128.185.120.209`

routing table-size *number-of-entries*

Sets the size of the router's IP routing table. The default size is 768 entries. Setting the routing table size too small causes dynamic routing information to be discarded. Setting the routing table size too large wastes router memory resources. See "Sizes" on page 365 for additional information about table sizes.

Valid Values: an integer number of entries in the range 64 to 65535

Default Value: 768 entries

Example: `set routing table-size 1000`

tag Configures the per-interface tags associated with received RIP information. These tags can be used to group routes together for later readvertisement via BGP where a tag will be treated as if it were a route's source autonomous system (AS) number. (Refer to the information on originate, send, and receive policies in the chapter "Using and Configuring BGP" in *8371 Interface Configuration and Software User's Guide*.) Tags are propagated also by the OSPF routing protocol.

Valid Values: an integer in the range 0 to 65535

Default Value: 0

Example: `set tag`

```
Interface address [0.0.0.0]? 1.1.1.1
Interface tag (AS number) [0]? 1
```

ttl Specifies the time-to-live for packets originated by the router.

Valid Values: a numeric in the range 1 to 255

Default Value: 64

Example: `set ttl 255`

Update

Use the **update packet-filter** command at the `IP config>` prompt to assign access control entries. The router prompts you for the name of the filter that you want to update. The `IP config>` prompt changes to incorporate the packet filter name you provide.

Syntax:

update *packet-filter-name*

packet-filter-name

Specifies the name of the packet filter to be updated. You must have created that filter using the **add packet-filter** command and you must use the **set access-control** command to enable the packet filter.

Valid Values: any 16-character name.

You can include dashes (-) and underscores (_) in the name.

Default Value:none

```
IP config> update packet-filter
Packet-filter name [ ]? pf-1-in
Packet-filter 'pf-1-in' Config>
```

You can access a list of sub-commands by typing **?** at the `Packet-filter 'name' Config>` prompt.

```
Packet-filter 'test' Config> ?
LIST
CHANGE
DELETE
ADD
MOVE
EXIT
```

Adding and Changing Access Controls to a Packet Filter

Use the **add access-control** command to add access controls to the specified packet filter. The router prompts you for the access control type (either Exclusive or Inclusive), and the source and destination addresses and masks of packets to which the filter will apply.

type Indicates what is done with packets that match the access control rule parameters.

E Exclusive; matching packets are discarded.

I Inclusive; matching packets are processed further by the router.

Default Value: Exclusive

source address
Valid Values: A valid IP address in dotted decimal notation.

Default Value: 0.0.0.0

source mask
Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: 255.255.255.255

destination address
Valid Values: A valid IP address in dotted decimal notation.

Default Value: 0.0.0.0

destination mask
Valid Values: 0.0.0.0 to 255.255.255.255

Default Value: 255.255.255.255

first protocol
The lower boundary of a protocol number range.

The commonly used protocol numbers are:
- **1** for ICMP
- **6** for TCP
- **8** for EGP
- **17** for UDP
- **89** for OSPF.

See RFC 1340, “Assigned Numbers” for details on IP protocol numbers.

Valid Values: 0 to 255

Default Value: 0

last protocol
The upper boundary of a protocol number range.

The commonly used protocol numbers are:
- **1** for ICMP
- **6** for TCP
- **8** for EGP
- **17** for UDP
- **89** for OSPF.

See RFC 1340, “Assigned Numbers” for details on IP protocol numbers.

Valid Values: 0 to 255

Default Value:0

first port

The lower boundary of a IP TCP/UDP port range.

Valid Values: a port number in the range 0 to 65535

Address Default Value: 0

Some commonly used port numbers are:

- 21 for FTP
- 23 for Telnet
- 25 for SMTP
- 513 for rlogin
- 520 for RIP

last port

The upper boundary of a IP TCP/UDP port range.

Valid Values: a port number in the range 0 to 65535

Address Default Value: 0

Some commonly used port numbers are:

21 for FTP
23 for Telnet
25 for SMTP
513 for rlogin
520 for RIP

Example: This example of the **add access-control** command shows how to exclude all incoming packets originating from network 128.185.0.0 and received on interface 0.

```
Packet-filter 'pf-in-0' Config> add access-control
Enter type [E]?
Internet source [0.0.0.0]? 128.185.0.0
Source mask [255.255.255.255]? 255.255.0.0
Internet destination [0.0.0.0]?
Destination mask [255.255.255.255]? 0.0.0.0
Enter starting protocol number ([CR] for all) [-1]?
```

Use the **change access-control** command to change existing access controls using the index number of the access control that you want to change.

You can use the **list access-control** command to view the access controls configured for each packet filter.

```
Packet-filter 'pf-in-0' Config> list access-control
Access Control is: enabled
List of access control records:

                                                    Beg  End Beg End
    Ty Source      Mask      Destination  Mask     Pro  Pro Prt Prt
1   E  128.185.0.0 FFFF0000  0.0.0.0      00000000 0   255  0  65535
2   I  0.0.0.0     00000000  0.0.0.0      00000000 0   255  0  65535
```

You can change the order of a packet filter's access control records with the **move access-control** command as shown.

```
Packet-filter 'test' Config> move access-control
Enter index of control to move [1]?
Move record AFTER record number [0]? 2
About to move:

                                                  Beg  End Beg End
```

```
    Ty Source      Mask      Destination  Mask    Pro  Pro Prt Prt
1   E  10.0.0.0    FFFF0000  0.0.0.0      00000000  0  255  0  65535
to be after:
2   I  10.5.5.0    FFFF0000  1.2.3.4      FF0000FF 0   255  0  65535
Are you sure this is what you want to do (Yes or [No]): y
```

Deleting Access Controls for a Filter

Use the **delete access-control** command to delete a record from a packet filter's access-control list.

```
Packet-filter 'test' Config> delete access-control
Enter index of access control to be deleted [1]? 4
```

The router responds by displaying the access-control record you have specified.

```
                                                  Beg  End Beg End
    Ty Source      Mask      Destination  Mask    Pro  Pro Prt Prt
4   I  1.2.9.9     FF0000FF  0.0.0.0      00000000 0   255  0  65535
Are you sure this is the record you want to delete (Yes or [No]): y
Deleted
Packet-filter 'test' Config>
```

Exiting the Access Controls Process

Exit the access controls process by typing **exit** at the prompt. This returns you to the `IP config>` prompt.

```
Packet-filter 'test' Config> exit
IP config>
```

For the **disable** and **enable** commands, the keyword **source-addr-verification** can be configured only from the `Packet-filter '`*`filter-name`*`' Config>` prompt.

Route Filter Policy Configuration

This section describes the subset of commands used to configure route filter policies. To access this subset of IP configuration commands, follow these steps:

1. Create a route filter policy. See the **add route-policy** command on page 316.
2. Use the **change route-policy** command to bring up the `IP Route Policy Config>` prompt. The `IP Route Policy Config>` prompt applies only to the particular route policy identified by the **change route-policy** command.

   Example:

```
IP config>change route-policy ospf-import
ospf-import IP Route Policy Configuration
IP Route Policy Config>
```

Note: Route filter policies can be used to determine which routes are imported in OSPF and the specific details of their advertisement, including OSPF external type, metric, and tag value. Refer to the **enable as boundary routing** command on page 467 for information about using route filter policies to configure OSPF.

Route filter policies can also be used to control what routes are advertised or accepted when RIP is used. See the previously described **enable receiving**, **enable sending**, **disable receiving**, and **disable sending** commands.

Table 58. IP Route Policy Configuration Commands Summary

Command	Function
Add	Adds an action, an entry, or a match condition to a route filter policy.
Delete	Deletes an action, an entry, or a match condition from a route filter policy.

Table 58. IP Route Policy Configuration Commands Summary (continued)

Command	Function
List	Lists the route policy entries, actions, and match conditions for the route policy currently being changed.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **add** command to add route filter policy entries to the route filter policy, to add match conditions to existing entries, or to add actions to existing entries.

Syntax:

add action . . .
entry . . .
match-condition . . .

action . . .
Adds an action to an existing route filter policy entry. Adding an action to a route filter policy is optional. One action can be added to each entry. If you need more than one action to apply to one address or address range, specify a second entry for that address or range. Then, define the second action for the second entry. These are the actions that can be specified:

Syntax:

auto-tag
set manual-tag
set metric
set route-type

auto-tag *route-policy-index*
Automatically sets the tag for the route, using a routing protocol specific heuristic. This option is described in RFC 1745.

route-policy-index
Identifies the entry to which the action should be applied.

Valid Value: 1 to 65535

Default Value: none

set manual-tag *route-policy-index manual-tag*
Sets the manual tag for the route to the specified value. This tag is usually the AS number when the protocol is OSPF.

route-policy-index
Identifies the entry to which the action should be applied.

Valid Value: 1 to 65535

Default Value: none

manual-tag

Valid Value: X'0' to X'FFFFFFFF'

Default Value: none

set metric *route-policy-index metric*

Sets the metric for the route to the specified value.

route-policy-index

Identifies the entry to which the action should be applied.

Valid Value: 1 to 65535

Default Value: none

metric

Valid Value: 1 to 255

Default Value: none

set route-type *route-policy-index route-type*

Sets the OSPF external route type. This action is ignored for applications other than OSPF AS boundary route importation.

route-policy-index

Identifies the entry to which the action should be applied.

Valid Value: 1 to 65535

Default Value: none

route-type

Valid Value: 1 or 2

Default Value: none

entry *route-policy-index ip-address ip-mask address-match policy-type*

Adds a route filter policy entry to the route filter policy being changed. Each entry within a route filter policy is identified by a unique index number, which is manually configured. If the entry with the specified index number already exists, that entry is changed according to the new parameters configured.

When you add the route filter policy, you define the processing of the entries as either strictly linear or longest match. If the route filter policy processing is strictly linear, the route filter policy entries are processed according to the ascending order of their index numbers. If the route filter policy processing is longest match, the route filter policy entries are processed according to the IP address and mask that has the longest match. If multiple route filter policy entries have the same IP address and mask when longest match is used, then the match will be in order of ascending index number among the entries with the same IP address and mask.

route-policy-index

Identifies the entry.

Valid Value: 1 to 65535

Default Value: none

ip-address

Valid Value: any valid IP address

Default Value: none

ip-mask

Valid Value: any valid IP mask

Default Value: none

address-match exact/range

If this value is *exact*, the route filter policy entry will match only on a route with that exact address and mask. If this value is *range*, the route filter policy entry will match on any route that is within the range encompassed by the address and mask, including the exact route.

Valid Value: exact or range

Default Value: range

policy-type inclusive/exclusive

If this value is *inclusive*, routes matching this route filter policy entry are included in the routing table. If this value is *exclusive*, routes matching this route filter policy entry are excluded, that is, they are not entered into the routing table. Even if actions are configured for a route filter policy entry that is exclusive, these actions are not applicable.

Valid Value: inclusive or exclusive

Default Value: inclusive

match-condition . . .

Adds a match condition to an existing route filter policy entry. A match condition, which is an optional parameter or set of parameters, is applied to a route that the entry definition has matched. The match condition filters the packet for particular conditions in addition to the IP address and the IP mask. Only one match condition can be configured per entry. To use two match conditions for the same address or address range, you can add a second entry to the route filter policy and specify the second match condition for that entry. These are the match conditions:

Syntax:

as

gateway

metric

net

protocol

source-gateway

as *route-policy-index as-number*

Matches the route according to its AS number. This value is interpreted only when the route filter policy is applied to AS boundary routing.

route-policy-index

An integer that identifies the entry with which the match should be made.

Valid Value: 1 to 65535

Default Value: none

as-number

Valid Value: 1 to 65535

Default Value: none

gateway *route-policy-index gateway-address-and-mask*
Matches the route with a next-hop gateway in the specified range.

route-policy-index
Identifies the entry with which the match should be made.

Valid Value: 1 to 65535

Default Value: none

gateway-address-and-mask

Valid Value: a valid IP address and mask

Default Value: none

metric *route-policy-index lower-metric-number upper-metric-number*
Matches the metric of the route with the one of the numbers in a range of metric numbers. You will be prompted for two numbers to identify the range of metric numbers: one for the low end of the range and one for the high end. If you want a single metric number, specify the same number twice.

route-policy-index

Valid Value: 1 to 65535

Default Value: none

lower-metric-number

Valid Value: 1 to 65535

Default Value: none

upper-metric-number

Valid Value: 1 to 65535

Default Value: none

net *route-policy-index lower-net-number upper-net-number*
Matches the routes that have a next hop with an outgoing network number in the range defined by the lower and upper network numbers. You will be prompted for two numbers to identify the range of outgoing network numbers: one for the low end of the range and one for the high end. If you want a single network number, specify the same number twice.

route-policy-index
Identifies the entry with which the match should be made.

Valid Value: 1 to 65535

Default Value: none

lower-net-number
The lower bound of the network number range for matching next-hop outgoing networks. These can be viewed using the **list devices** command from the `Config>` prompt.

Valid Value: 1 to 65536

Default Value: none

upper-net-number
The upper bound of the network number range for matching next-hop outgoing networks.

Valid Value: 1 to 65536

Default Value: none

protocol *protocol route-policy-index*
Matches the route with a protocol.

protocol

Valid Values:

Syntax:

bgp
direct
natural-nets
ospf-intra
ospf-inter
ospf
ospf-all
ospf-ext
ospf-e1
ospf-e2
rip
static

Default Value: none

route-policy-index
An integer that identifies the entry with which the match should be made.

Valid Value: 1 to 65535

Default Value: none

source-gateway *route-policy-index ip-address-and-mask*
Matches routes that come from a specified source gateway or a range of source gateways.

route-policy-index
An integer that identifies the entry with which the match should be made.

Valid Value: 1 to 65535

Default Value: none

ip-address-and-mask

Valid Values: any valid IP address and mask combination

Default Value: none

Delete

Use the **delete** command to delete route filter policy entries, match-conditions from existing route filter policy entries, or actions from existing route filter policy entries. See the **add** command in this section for a description of the parameters that can be deleted.

List

Use the **list** command to list the route filter policy entries, match conditions, and actions that exist for the route filter policy currently being changed.

Syntax: list

Example:

```
IP Route Policy Config>list

     IP Address      IP Mask           Match  Index  Type
     ------------------------------------------------------
     9.0.0.0         255.0.0.0         Range  1      Include
     10.0.0.0        255.0.0.0         Range  2      Exclude
         Match Conditions:  Protocol: BGP
     0.0.0.0         0.0.0.0           Range  3      Include
         Match Conditions:  Protocol: Static
                            Gateway IP Address Range: 153.2.2.20/255.255.255.255
     10.1.1.0        255.255.255.0     Range  4      Include
     0.0.0.0         0.0.0.0           Range  7      Include
         Policy Actions:    Set Manual Tag: 0xACEEACEE
     0.0.0.0         0.0.0.0           Range  8      Include
         Match Conditions:  Protocol: RIP
```

Accessing the IP Monitoring Environment

Use the following procedure to access the IP monitoring commands. This process gives you access to the IP *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. For example:

   ```
   * talk 5
   +
   ```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the + prompt, enter the **protocol ip** command to get you to the IP> prompt.

   Example:

   ```
   + prot ip
   IP>
   ```

IP Monitoring Commands

This section describes the IP monitoring commands. Table 59 on page 356 lists the IP monitoring commands. The commands allow you to monitor the router's IP forwarding process. The monitoring capabilities include the following: configured parameters such as interface address and static routes can be viewed, the current state of the IP routing table can be displayed, and a count of IP routing errors can be listed.

IP Monitoring Commands (Talk 5)

Table 59. IP Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Access controls	List the current IP access control mode, together with the configured access control records.
Cache	Displays a table of all recent routed destinations.
Counters	Lists various IP statistics, including counts of routing errors and packets dropped.
Distributed IP Gateway	Lists whether a distributed gateway exists and if it is active or inactive.
Dump routing tables	Lists the contents of the IP routing table.
Interface addresses	Lists the router’s IP interface addresses.
Packet-filter	Displays the access-control information defined for the specified packet-filter, or all filters.
Parameters	Lists various parameter values.
Ping	Sends ICMP Echo Requests to another host and watches for a response. This command can be used to isolate trouble in an internetwork environment.
Redundant Default Gateway	Lists whether a redundant default gateway exists and if it is active or inactive.
Reset	Allows you to dynamically reset the IP/RIP configuration.
RIP	Displays the status of the RIP protocol.
RIP-Policy	Displays the route filter policy applied on the specified interface.
Route	Lists whether a route exists for a specific IP destination, and if so, the routing table entry that corresponds to the route.
Route-table-filtering	Lists any defined route filters and indicates whether route-filtering is enabled or disabled.
Sizes	Displays the size of specific IP parameters.
Static routes	Displays the static routes that have been configured. This includes the default gateway.
Traceroute	Displays the complete path (hop-by-hop) to a particular destination.
UDP-Forwarding	Displays the UDP port numbers and destination IP addresses that you added using the **add** command or the **enable** command.
VRID	Displays detailed information for a specific VRID
VRRP	Lists the summary status for the VRRP protocol.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Access Controls

Use the **access controls** command to print the global access control mode in use together with a list of the configured global access control rules.

Access control is either disabled (meaning that no access control is being done and the access control rules are being ignored) or enabled (meaning that access control is being done and the access control rules are being recognized). The **set access on** talk 6 command enables access control.

Syntax:

access

Example: access

```
           Access Control currently enabled
           Access Control facility: USER
           Access Control run 702469 times, 657159 cache hits

           List of access control records:



2  Type=E    Source=0.0.0.0          Dest=0.0.0.0           Prot= 1
             SMask =255.255.255.255  DMask=255.255.255.255  Use=18962
             Sports=  N/A            Dports=  N/A
                      T/C= 1/**      Log=Yes ELS=N SNMP=N   SLOG=L(Alert)

3  Type=I    Source=1.1.1.1          Dest=1.1.1.2           Prot= 6
             SMask =255.255.255.255  DMask=255.255.255.254  Use=42
             Sports=  2-200          Dports=  1-100
                                     Log=No

4  Type=I    Source=9.1.2.3          Dest=0.0.0.0       Prot=  0-255
             SMask =255.255.255.255  DMask=0.0.0.0      Use=0
             SPorts=  0-65535        DPorts=  0-65535
                      T/C= **/**     Log=N
             Tos=xE0/x00-x00         ModifyTos=x1F/x08
             PbrGw=9.2.160.1         UseDefRte=Y

5  Type=I    Source=0.0.0.0        Dest=0.0.0.0         Prot= 0-255
             Mask=0.0.0.0          Mask=0.0.0.0         Use=683194
             Sports=  1-65535      Dports=  1-65535
                                   Log=No
```

Exclusive (E) means that packets matching the access control rule are discarded. Inclusive (I) means that packets matching the access control rule are forwarded. When access control is enabled, packets failing to match any access control record are discarded. *Prot* (protocol) indicates the IP protocol number. *Sports* indicates the range of TCP/UDP source port numbers; *Dports* indicates the range of TCP/UDP destination port numbers. *SYN* indicates TCP connection establishment filtering. *T/C* stands for ICMP type and code; *SLOG* stands for SysLog.

The Use field specifies the number of times the access control system matched a particular record to an incoming packet, for example, the number of times that a particular record in the IP access controls system was invoked by the characteristics of an incoming or outgoing packet.

In this example, access control rule number 4 has activated the TOS filter. The TOS parameters are shown. See the **add access-control** command in talk 6 for a description of these parameters.

Cache

Use the **cache** command to display the IP routing cache, which contains recently routed destinations. If a destination is not in the cache, the router looks up the destination in the routing information table in order to make a forwarding decision.

Syntax:

cache

Example: cache

```
           Destination     Usage          Next hop
           128.185.128.225 1              128.185.138.180 (Eth/0)
           192.26.100.42   1              128.185.138.180 (Eth/0)
           128.185.121.1   18             128.185.123.18  (PPP/0)
           128.185.129.219 76             128.185.125.25  (PPP/1)
           128.185.129.41  130            128.185.125.25  (PPP/1)
```

```
128.185.129.134 546            128.185.125.40  (PPP/1)
128.185.129.221 1895           128.185.125.40  (PPP/1)
128.185.129.193 96             128.185.125.40  (PPP/1)
128.197.3.4     4              128.185.123.18  (PPP/0)
128.185.128.25  98             128.185.125.41  (PPP/1)
128.185.124.121 4              128.185.124.121 (Eth/0)
128.185.136.203 95             128.185.125.39  (PPP/1)
128.185.194.4   581            128.185.125.39  (PPP/1)
128.185.123.17  2              128.185.123.17  (PPP/0)
192.26.100.42   1              128.185.125.38  (PPP/1)
128.52.22.6     2              128.185.123.18  (PPP/0)
128.197.3.2     1              128.185.123.18  (PPP/0)
128.185.126.24  61             128.185.125.25  (PPP/1)
128.185.138.150 482            128.185.125.39  (PPP/1)
128.185.123.18  152            128.185.123.18  (PPP/0)
```

Destination
IP destination host.

Usage Number of packets recently sent to the destination host.

Next hop
IP address of the next router on the path toward the destination host. Also displayed is the network name of the interface used by the sending router to forward the packet.

Counters

Use the **counters** command to display the statistics related to the IP forwarding process. This includes a count of routing errors, along with the number of packets that have been dropped due to congestion.

Syntax:

counters

Example: `counters`

```
Routing errors
Count   Type
    0   Routing table overflow
 2539   Net unreachable
    0   Bad subnet number
    0   Bad net number
    0   Unhandled broadcast

    0   Unhandled directed broadcast
 4048   Attempted forward of LL broadcast

Packets discarded through filter  0
IP multicasts accepted:           60592

 IP input packet overflows
    Net   Count
   Eth/0  0
    FR/0  0
```

Routing table overflow
Lists the number of routes that have been discarded due to the routing table being full.

Net unreachable
Indicates the number of packets that could not be forwarded due to unknown destinations. This does not count the number of packets that have been forwarded to the authoritative router (default gateway).

Bad subnet number
Counts the number of packets or routes that have been received for illegal subnets (all ones or all zeros).

Bad net number
Counts the number of packets or routes that have been received for illegal IP destinations (for example, class E addresses).

Unhandled broadcasts
Counts the number of (non-local) IP broadcasts received (these are not forwarded).

Unhandled multicasts
Counts the number of IP multicasts that have been received, but whose addresses were not recognized by the router (these are discarded).

Unhandled directed broadcasts
Counts the number of directed (non-local) IP broadcasts received when forwarding of these packets is disabled.

Attempted forward of LL broadcast
Counts the number of packets that are received having non-local IP addresses but were sent to a link–level broadcast address. These are discarded.

Packets discarded through filter
Counts the number of received packets that have been addressed to filtered networks/subnets. These are discarded silently.

IP multicasts accepted
Counts the number of IP multicasts that have been received and successfully processed by the router.

IP packet overflows
Counts the number of packets that have been discarded due to congestion at the forwarder's input queue. These counts are sorted by the receiving interface.

Distributed IP Gateway

Use the **distributed ip gateway** command to display the distributed IP Gateways configured for each interface.

Syntax:

distributed ip gateway

Example

```
Distributed IP Gateways for each interface:
   inf  3   22.2.2.6   255.0.0.0   00.00.00.00.00.AB   backup standby
   inf  4   11.1.1.6   255.0.0.0   00.00.00.00.00.BA   primary active
```

Note: Type can be "Primary" or "Backup". Status can be "Active" or "Standby".

Dump Routing Table

Use the **dump** command to display the IP routing table. A separate entry is printed for each reachable IP network/subnet. The IP default gateway in use (if any) is listed at the end of the display.

Syntax:

dump

Example: `dump`

```
Type   Dest net         Mask     Cost Age   Next hop(s)
SPE1   0.0.0.0          00000000  4    3    128.185.138.39  (2)
SPF*   128.185.138.0    FFFFFF00  1    1    Eth/0
Sbnt   128.185.0.0      FFFF0000  1    0    None
SPF    128.185.123.0    FFFFFF00  3    3    128.185.138.39  (2)
SPF    128.185.124.0    FFFFFF00  3    3    128.185.138.39  (2)
SPF    192.26.100.0     FFFFFF00  3    3    128.185.131.10  (2)
RIP    197.3.2.0        FFFFFF00  10   30   128.185.131.10
RIP    192.9.3.0        FFFFFF00  4    30   128.185.138.21
Del    128.185.195.0    FFFFFF00  16   270  None


Default gateway in use.

Type Cost Age  Next hop
SPE1 4    3    128.185.138.39

Routing table size: 768 nets (36864 bytes), 36 nets known
```

Type Indicates how the route was derived.

Sbnt - Indicates that the network is subnetted; such an entry is a place-holder only.

Dir - Indicates a directly connected network or subnet.

RIP - Indicates that the route was learned through the RIP protocol.

Del - Indicates that the route has been deleted.

Stat - Indicates a statically configured route.

BGP - Indicates routes learned through the BGP protocol.

BGPR - Indicates routes learned through the BGP protocol that are readvertised by OSPF and RIP.

Fltr - Indicates a routing filter.

SPF - Indicates that the route is an OSPF intra-area route.

SPIA - Indicates that it is an OSPF inter-area route.

SPE1, SPE2 - Indicates OSPF external routes (type 1 and 2 respectively)

Rnge - Indicates a route type that is an active OSPF area address range and is not used in forwarding packets.

Dest net
IP destination network/subnet.

Mask IP address mask.

Cost Route Cost.

Age For RIP and BGP routes, the time that has elapsed since the routing table entry was last refreshed.

Next Hop
IP address of the next router on the path toward the destination host. Also displayed is the interface type used by the sending router to forward the packet.

An asterisk (*) after the route type indicates that the route has a static or directly connected backup. A percent sign (%) after the route type indicates that RIP updates will always be accepted for this network/subnet.

A number in parentheses at the end of the column indicates the number of equal-cost routes to the destination. The first hops belonging to these routes can be displayed with the IP **route** command.

Interface Addresses

Use the **interface addresses** command to display the router's IP interface addresses. Each address is listed together with its corresponding hardware interface and IP address mask.

Hardware interfaces having no configured IP interface addresses will not be used by the IP forwarding process; they are listed as Not an IN net. There is one exception.

Syntax:

interface

Interface
Indicates the hardware type of the interface.

IP addresses
Indicates the IP address of the interface.

Mask Indicates the subnet mask of the interface.

Packet-filter

Use the **packet-filter** command to display information defined for a specific packet filter, or for all filters. Packet-filters are interface-specific lists of access control records.

Syntax: packet-filter [*name*]

Example: `packet-filter pf-in-0`

```
Name              Direction     Interface   #Access-Controls
pf-in-0           In            0           2

Access Control currently enabled
Access Control run 8 times, 7 cache hits

List of access control records:

                                                    Beg End  Beg  End
  Ty  Source       Mask     Destination  Mask      PPP PPP  Port Port  Use
0 I   0.0.0.0      00000000 192.67.67.20 00000000  6   6    25   25    0
1 E   150.150.1.0  FFFFFF00 150.150.2.0  00000000  0   255  0    655   0
2 I   0.0.0.0      00000000 0.0.0.0      00000000  89  89   0    655   27
```

Parameters

Use the **parameters** command to list the values of various parameters.

Example:

```
IP> parameters
ARP-SUBNET-ROUTING     : disabled
ARP-NET-ROUTING        : disabled
CLASSLESS              : disabled
DIRECTED-BROADCAST     : enabled
ECHO-REPLY             : enabled
FRAGMENT-OFFSET-CHECK  : disabled
REASSEMBLY-SIZE        : 12000 bytes
RECORD-ROUTE           : enabled
ROUTING TABLE-SIZE     : 768 entries (52224 bytes)
(Routing) CACHE-SIZE   : 64 entries
SAME-SUBNET            : disabled
SOURCE-ROUTING         : enabled
TIMESTAMP              : enabled
```

```
TTL                    : 64

IP>
```

Ping

Use the **ping** command to have the router send ICMP Echo messages to a given destination (that is, “pinging”) and watch for a response. This command can be used to isolate trouble in the internetwork.

Syntax:

ping *dest-addr [src-addr data-size ttl rate tos data-value]*

The ping process is done continuously, incrementing the ICMP sequence number with each additional packet. Each matching received ICMP Echo response is reported with its sequence number and the round-trip time. The granularity (time resolution) of the round-trip time calculation is usually around 20 milliseconds, depending on the platform.

To stop the ping process, type any character at the console. At that time, a summary of packet loss, round-trip time, and number of unreachable ICMP destinations will be displayed.

When a broadcast or multicast address is given as destination, there may be multiple responses printed for each packet sent, one for each group member. Each returned response is displayed with the source address of the responder.

You can specify the size of the ping (number of data bytes in the ICMP message, excluding the ICMP header), value of the data, time-to-live (TTL) value, rate of pinging, and TOS bits to set. You can also specify the source IP address. If you do not specify the source IP address, the router uses its local address on the outgoing interface to the specified destination. If you are validating connectivity from any of the router’s other interfaces to the destination, enter the IP address for that interface as the source address.

Only the destination parameter is required; all other parameters are optional. By default the size is 56 bytes, the TTL is 64, the rate is 1 ping per second, and the TOS setting is 0. The first 4 bytes of the ICMP data are used for a timestamp. By default the remaining data is a series of bytes with values that are incremented by 1, starting at X'04', and rolling over from X'FF' to X'00' (for example, X'04 05 06 07 . . . FC FD FE FF 00 01 02 03 . . .'). These values are incremented only when the default is used; if the data byte value is specified, all of the ICMP data (except for the first 4 bytes) is set to that value and that value is not incremented. For example, if you set the data byte value to X'FF', the ICMP data is a series of bytes with the value X'FF FF FF . . .'.

Example:

```
IP> ping
Destination IP address [0.0.0.0]? 192.9.200.1
Source IP address [192.9.200.77]?
Ping data size in bytes [56]?
Ping TTL [64]?
Ping rate in seconds [1]?
Ping TOS (00-FF) [0]? e0
Ping data byte value (00-FF) [ ]?
PING 192.9.200.77-> 192.9.200.1:56 data bytes,ttl=64,every 1 sec.
56 data bytes from 192.9.200.1:icmp_seq=0.ttl=255.time=0.ms
56 data bytes from 192.9.200.1:icmp_seq=1.ttl=255.time=0.ms
```

```
   56 data bytes from 192.9.200.1:icmp_seq=2.ttl=255.time=0.ms


   ----192.9.200.1 PING Statistics----
   3 packets transmitted, 3 packets received, 0% packet loss
   round-trip min/avg/max=0/0/0 ms
   IP>
IP>ping
```

Redundant Default Gateway

Use the **redundant default gateway** command to display the redundant Default IP Gateways configured for each interface.

Syntax:

redundant default gateway

Example:

```
Redundant Default IP Gateways for each interface:
   inf  3   22.2.2.6   255.0.0.0   00.00.00.00.00.AB   backup standby
   inf  4   11.1.1.6   255.0.0.0   00.00.00.00.00.BA   primary active
```

Note: Type can be “Primary” or “Backup”. Status can be “Active” or “Standby”.

Reset IP

Use the **reset IP** command to make effective certain IP and RIP configuration changes. See “Response to IP Configuration Commands” on page 308 for a list of configuration changes made effective by this command.

Syntax:

reset ip

Example:

```
IP>interface
Interface  IP Address(es)   Mask(s)
  Eth/0    30.1.1.2         255.255.255.0
           30.1.1.1         255.255.255.0
           153.2.2.25       255.255.255.240
 IP>
*talk 6

IP config>add address 0 5.1.1.1 255.255.0.0
IP config>
*talk 5

IP>reset ip

IP>interface
Interface  IP Address(es)   Mask(s)
  Eth/0    5.1.1.1          255.255.0.0
           30.1.1.2         255.255.255.0
           30.1.1.1         255.255.255.0
           153.2.2.25       255.255.255.240
 IP>
```

RIP

Use the **rip** command to display the RIP protocol status detail.

Syntax:

rip

Example:

```
IP>rip

                                RIP Interfaces

Interface-Addr  Interface-Mask  Version  In Out  Send-Flags      Receive-Flags
10.69.1.2       255.255.255.0      1      1   0  D,P
200.1.1.2       255.255.255.0      2      1   0  Policy,P        Policy
Send Flags: N=Network S=Subnet H=Host St=Static D=Default O=Outage-Only
            P=PoisonReverse Policy=Send-Policy
Recv Flags: N=Network S=Subnet H=Host OSt=Override-Static OD=Override-Default
            Policy=Receive-Policy

        RIP Policy

Interface-Address  Send Policy        Receive-Policy
10.69.1.2          rip-global-send    rip-global-recv
200.1.1.2          rip-send           rip-receive
RIP global receive policy: rip-global-recv
RIP global send policy: rip-global-send

RIP never originates a default route
```

RIP-Policy

Use the **rip-policy** command to display the RIP policy that is currently applicable to the specified interface.

Syntax:

rip-policy

Example:

```
IP>rip-policy
For which interface [0.0.0.0]? 200.1.1.2

Interface Send Policy: rip-send for 200.1.1.2
Checksum 0x8637  Longest-Match Application

IP Address      IP Mask           Match  Index  Type
-----------------------------------------------------
0.0.0.0         0.0.0.0           Range  1      Include
    Match Conditions:  Protocol: BGP
    Policy Actions:    Set Manual Tag: 0xACEEACEE
                       Set Metric: 3

Interface Receive Policy: rip-receive for 200.1.1.2
Checksum 0x5049  Longest-Match Application

IP Address      IP Mask           Match  Index  Type
-----------------------------------------------------
0.0.0.0         0.0.0.0           Range  1      Include
    Match Conditions:  Source Gateway IP Address Range: 200.1.1.1/255.255.255.255
```

Route

Use the **route** command to display the route (if one exists) to a given IP destination. If a route exists, the IP addresses of the next hops are displayed, along with detailed information concerning the matching routing table entry. (See the IP **dump** command.)

Syntax:

route *ip-destination*

Example: `route 133.1.167.2`

```
Destination:    133.1.166.0
Mask:           255.255.254.0
Route type:     SPF
Distance:       1
Age:            1
Tag:            0
Next hop(s):    133.1.167.2       (FR/0)
```

Example: `route 128.185.230.0`

```
Destination:   128.185.230.0
Mask:          255.255.255.0
Route type:    SPF
Distance:      1
Age:           1
Next hop(s):   128.185.230.0    (TKR/0)
```

Example: `route 128.185.232.0`

```
Destination:   128.185.232.0
Mask:          255.255.255.0
Route type:    RIP
Distance:      3
Age:           0
Next hop(s):   128.185.146.4   (Eth/0)
```

Route-table-filtering

Use the **route-table-filtering** command to display whether or not route table filtering is enabled and list any defined route table filters.

Syntax:

route-table-filtering

Example: `route-table-filtering`

```
IP>route-table-filtering
Route Filters

Destination     Mask            Match Type
10.1.1.0        255.255.255.0   BOTH  E
10.1.1.1        255.255.255.255 EXACT I
50.0.0.0        255.0.0.0       BOTH  E
50.50.0.0       255.255.0.0     BOTH  I

IP>
```

Sizes

Use the **sizes** command to display the configured sizes of specific IP parameters.

Syntax:

sizes

Example: `sizes`

```
Routing table size:         768
Table entries used:         3
Reassembly size:            12000
Largest reassembled pkt:    0
```

Routing table size

The configured number of entries that the routing table will maintain.

Table entries used
The number of entries used from the routing table. This number includes both active and inactive entries. The value displayed using the “dump” command as “xx nets known” is the number of active routing table entries. The configured routing table size should be large enough to maintain current active entries as well as other anticipated routing entries.

Reassembly buffer size
The configured size of the reassembly buffer that is used to reassemble fragmented IP packets.

Largest reassembled pkt
The largest IP packet that this router has had to reassemble.

Static Routes

Use the **static routes** command to display the list of configured static routes. Configured default gateways and default subnet gateways are also listed.

Each static route’s destination is specified by an address-mask pair. Default gateways appear as static routes to destination 0.0.0.0 with mask 0.0.0.0. Default subnet gateways also appear as static routes to the entire IP subnetted network.

The following example shows a configured default gateway, a configured default subnet gateway (assuming 128.185.0.0 is subnetted), and a static route to network 192.9.10.0.

Syntax:

static

```
IP>static routes
Net             Mask            Cost  Next hop
1.1.0.0         255.255.0.0     1     10.1.1.1        TKR/0
                                2     20.1.1.1        TKR/1
                                3     30.1.1.1        TKR/2
2.2.0.0         255.255.0.0     10    10.2.2.2        TKR/0
3.3.0.0         255.255.0.0     100   10.3.3.3        TKR/0
                                200   20.3.3.3        TKR/1

IP>
```

Net The destination address of the route.

Mask The destination mask of the route.

Cost The cost of using this route.

Next Hop
The next router a packet would pass through using this route.

Traceroute

Use the **traceroute** command to display the entire path to a given destination, hop by hop. For each successive hop, **traceroute** sends out a default of three probes and prints the IP address of the responder, together with the round-trip time associated with the response. If a particular probe receives no response, an asterisk is displayed. Each line in the display relates to this set of three probes, with the left-most number indicating the distance from the router executing the command (in router hops).

The traceroute is done whenever the destination is reached, an ICMP Destination Unreachable is received, or the path length reaches a default maximum of 32 router hops.

When a probe receives an unexpected result, several indications can be displayed. "!N" indicates that an ICMP Destination Unreachable (net unreachable) has been received. "!H" indicates that an ICMP Destination Unreachable (host unreachable) has been received. "!P" indicates that an ICMP Destination Unreachable (protocol unreachable) has been received; because the probe is a UDP packet sent to a strange port, a port unreachable is expected. "!" indicates that the destination has been reached, but the reply sent by the destination has been received with a TTL of 1. This usually indicates an error in the destination, prevalent in some versions of UNIX, whereby the destination is inserting the probe's TTL in its replies. This unfortunately leads to a number of lines consisting solely of asterisks before the destination is finally reached.

Syntax:

traceroute *dest-addr [src-addr data-size probes wait tos max-ttl]*

dest-addr
The address at the far end of the route.

src-addr
The source address from which the trace originates.

data-size
The size in bytes of the data field of the traceroute message. The data field does not include the UDP header.

probes
Number of UDP traceroute messages sent from each hop.

wait Time in seconds between retries.

tos The setting of the TOS bits in the UDP messages. For example, a value of X'10' (B'00010000') sets the TOS bits to B'1000'. The default is 0, which sets the TOS bits to B'1000'.

max-ttl
Maximum time-to-live in seconds for each message.

Example:

```
IP> traceroute
Destination IP address [0.0.0.0]? 128.185.142.239
Source IP address [128.185.142.1]?
Data size in bytes [56]?
Number of probes per hop [3]?
Wait time between retries in seconds [3]?
Maximum TTL [32]?
Traceroute TOS (00-FF) [0]? 10

TRACEROUTE 128.185.142.1 -> 128.185.142.239: 56 data bytes
1 128.185.142.7 16 ms 0 ms 0 ms
2 128.185.123.22 16 ms 0 ms 16 ms
3 * * *
4 * * *
5 128.185.124.110 16 ms ! 0 ms ! 0 ms !
```

TRACEROUTE
Displays the destination area address and the size of the packet being sent to that address.

1 The first trace showing the destination's NSAP and the amount of time it took the packet to arrive at the destination. The packet is traced three times.

Destination unreachable
Indicates that no route to destination is available.

3 * * * Indicates that the router is expecting some form of response from the destination, but the destination is not responding.

UDP-Forwarding

Use the **UDP-forwarding** command to display the UDP port and addresses that you added using the **add udp-destination** command or the **enable udp-forwarding** command.

Syntax:

udp-forwarding

Example: `udp-forwarding`

```
UDP Port   IP Address
    35      20.2.1.1
    20      22.2.1.2
```

VRRP

Use the **VRRP** command to display summary information

Syntax:

vrrp

Example:

```
                                --VRID Summary--
IP address       VRID  State  Advertise Master-Dead  Address(es)
153.2.2.25          1  MASTER         1         N/A  153.2.2.25
                                                     5.1.1.1
```

Chapter 31. Using IPX

This chapter describes how to use the IPX protocol on your IBM 8371. It includes the following sections:

- "IPX Overview"
- "Configuring IPX" on page 370
- "Optional Configuration Tasks" on page 371

IPX Overview

IBM's implementation of IPX allows the router to function as a Novell NetWare internetwork router. It has these characteristics:

- Compatibility with all previous Novell NetWare version environments.
- Compatibility with the bridging function in a NetWare file server, plus a stand-alone NetWare bridge.
- Support for the Novell NetBIOS emulator.

IPX Addressing

The following sections describe IPX addressing.

Network Numbers

An IPX network number specifies the location of a particular network in an internetwork. You can use multi-part addresses like the city-street-house address on a piece of mail. For example, IPX refers to network numbers (city), host numbers (street), and socket numbers (house). These addresses allow communication between two entities on different networks.

Host Numbers

Each IPX circuit needs a 6-byte host (node) number.

Because serial lines have no hardware MAC addresses, you must specify a unique host number.

ATM circuits use their End System Identifier (ESI) as their host number. Their burned-in ESI will be used if one has not been configured.

IPX Circuits

The IPX routing software models network interfaces as either a single IPX broadcast circuit, as one or more IPXWAN point-to-point circuits, or a combination of both types of circuits. The type of encapsulation, IPX addressing and routing protocols used on the circuit depend on the underlying DLC and whether the IPX circuit is configured as broadcast or IPXWAN point-to-point.

IPX broadcast circuits have the following characteristics:

- Used on LAN interfaces
- Restricted to a single IPX broadcast circuit per interface
- Must be assigned a non-zero IPX network number

© Copyright IBM Corp. 1996, 1999

- For LANs it uses the network interface's MAC address as the circuit's IPX node number
- Allows concurrent use of RIP/SAP and Static routes and services.

The following sections describe the modeling of each type of supported network interfaces.

Native ATM

The IPX routing software models an ATM interface as a single IPX broadcast circuit. As such, the underlying ATM PVCs and SVCs defined by the user to interconnect routers on the ATM network are transparent to the IPX routing software.

The circuit must be assigned a unique IPX non-zero network number.

The ESI component of the ATM address is used as the circuit's IPX node number. The burned-in ESI will be used if an ESI has not been configured in the IPX ATM ARP-client associated with the ATM interface.

The LAN all-stations address (x'FFFFFFFFFFFF') serves as the IPX broadcast address. Packets addressed to the broadcast address are transmitted on all VCs on the interface by the underlying ATM DLC.

The IPX maximum packet size is derived from the MTU configured for the interface.

The underlying ATM DLC uses ATM InARP to map destination IPX node addresses to the appropriate ATM VC. Optionally, destination IPX node addresses can be statically configured for VCs connected to routers which do not support ATM InArp.

In order to support non-fully meshed ATM topologies, split-horizon can be disabled on the circuit. This allows RIP and SAP to propagate information to all VCs on the interface so that intermediate routing between VCs on the same interface can occur.

Fully-meshed ATM topologies need not disable split-horizon.

Any or all of the following routing types may be used on the circuit:

- Static Routes/Services
- RIP/SAP (Numbered)

Configuring IPX

This section describes how to initially configure IPX. The following sections describe optional parameters you can set.

1. Display the IPX configuration prompt as shown here:

```
* talk 6
Config> protocol ipx
IPX protocol user configuration
IPX config>
```

2. Enable IPX globally.

```
IPX config> enable ipx
```

3. Add a broadcast-circuit on LAN.

```
IPX Config>add broadcast-circuit
Which interface [0]? 1
IPX circuit number[3]? 5
IPX network number in hex
('0' is only allowed on IPXWAN unnumbered circuits) [1]? 01
```

Note: IPX network number 0 is valid only on IPXWAN unnumbered RIP or static routing circuits. IPX network number FFFFFFFF is not a valid IPX network number. IPX network number FFFFFFFE is reserved for the IPX Default Route and may not be used as an IPX network number.

4. Optionally, change the frame type for Ethernet or ATM LAN Emulation Client. You do not have to set the frame type for circuits other than Ethernet or ATM LAN Emulation Client. See “Frame” on page 397 for a description of available frame types.

   The default encapsulation formats are:

   - Ethernet - Ethernet_8023
   - Token Ring - Token-ring MSB

   Use the **frame** command as shown here:

```
IPX config> frame ethernet_8023
IPX circuit number [1]? 2
```

Optional Configuration Tasks

Optional settings that you can adjust are described in the following sections.

- “Specifying the Size of IPX RIP Network Table”
- “Specifying RIP Update Interval” on page 372
- “Specifying the Size of IPX SAP Services Table” on page 372
- “Specifying SAP Update Interval” on page 372
- “IPX Keepalive and Serialization Packet Filtering” on page 373
- “Configuring Multiple Routes” on page 373
- “Configuring Static Routes” on page 374
- “Configuring Static Services” on page 374
- “Configuring the RIP Default Route” on page 375
- “Configuring Global IPX Filters (IPX Access Controls)” on page 376
- “Global SAP Filters” on page 378
- “IPX Circuit Filters - Overview” on page 379
- “IPX Performance Tuning” on page 381
- “Split-Horizon Routing” on page 383

Specifying the Size of IPX RIP Network Table

The IPX RIP network table contains information about each IPX network. The default table size is 32. You can configure the table size from 1 to 2048; however, there may be memory limitations on the router that can prevent the maximum table size from being used.

```
IPX config>set maximum networks
New Network table size [32]? 32
```

Specifying RIP Update Interval

IPX uses RIP to maintain routes in its routing tables. A route indicates the path a packet follows. The RIP update interval determines how often the router broadcasts its routing information tables to its circuits. It also determines how long a RIP entry remains before being aged-out.

Valid entries remain in the routing tables for a period of three multiples of the RIP update interval, and the router broadcasts its RIP tables once every update interval.

For example, the default interval is 1 minute, which allows a valid entry to remain in the table for 3 minutes. After this time, if an entry is not refreshed by a RIP update, the route is marked with a hop count of infinity (16) and then it is deleted. Every 60 seconds the router broadcasts its RIP tables to corresponding circuits.

You can configure the RIP interval from 1 to 1440 minutes (24 hours). Increasing the RIP interval reduces traffic on WAN lines and dial circuits. It also prevents dial-on-demand circuits from dialing out as often.

Note: While complete RIP advertisements are controlled by the interval, the router still propagates network topology changes as quickly as it learns them.

The RIP interval is not configurable on the Novell file server.

```
IPX config>set rip-update-interval
IPX circuit number [1]? 2
RIP timer value(minutes) [1]? 2
```

Specifying the Size of IPX SAP Services Table

The IPX Service Advertising Protocol (SAP) services table is a distributed database used to find NetWare Services, such as file servers. Services are uniquely identified by a 2-byte numeric type and a 47-character name. Each service provider advertises its services, specifying service type, name, and address. The router accumulates this information in a table and sends it to other routers. The default table size is 32.

You can configure the table size from 1 to 2048; router memory constraints may prevent the maximum table size from being used.

```
IPX config>set maximum services
New Service table size [32]? 32
```

Specifying SAP Update Interval

The IPX Service Advertising Protocol (SAP) interval lets you configure the time between IPX SAP updates on a per-circuit basis. All router circuits on the same network must use the same SAP interval. This interval determines both the age-out time for table information, and the interval between broadcasts to router circuits.

Valid entries remain in the SAP services table for a period of three multiples of the SAP update interval, and the router broadcasts its SAP services table information once every update interval.

You can configure the SAP interval from 1 to 1440 minutes (24 hours). Increasing the SAP interval reduces traffic on WAN lines and dial circuits. It also prevents dial-on-demand circuits from dialing out as often.

Note: While complete SAP advertisements are controlled by this interval, the router still propagates network topology changes as quickly as it learns them.

The SAP interval is not configurable on the Novell file server.

```
IPX config>set sap-update
IPX circuit number [1]? 2
SAP timer value(minutes) [1]? 4
```

IPX Keepalive and Serialization Packet Filtering

You can configure IPX to prevent Keepalive and serialization packets from continually activating a dial-on-demand link or to minimize traffic over a dial-on-demand link.

In Figure 26, for example, if the Novell Client logs into the Novell Server and then remains idle, the server sends periodic Keepalive requests to the client and the client replies with Keepalive replies. Keepalive filtering causes the routers to enter the first Keepalive reply into their Keepalive tables and then forward the reply. After that, the routers do not forward Keepalive traffic for that client-server connection over the WAN link. Instead, Router A replies to Keepalive requests it receives from the server and Router B sends Keepalive requests to the Novell Client.

Keepalive filtering also prevents the routers from forwarding NetWare serialization packets over the WAN link.

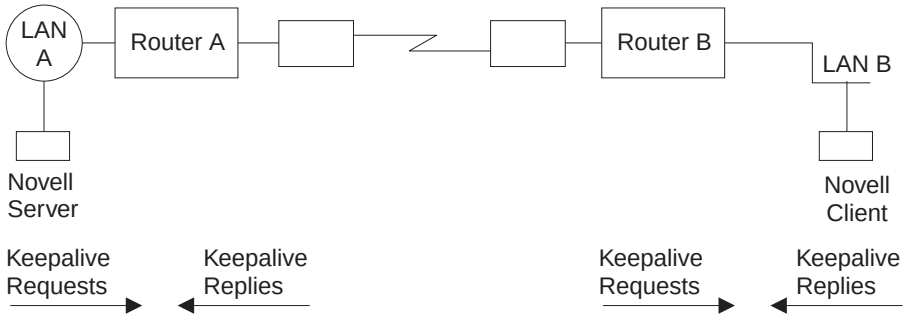

Figure 26. Keepalive Filtering

To set up Keepalive filtering, enable it on the dial circuits.

```
IPX Config> enable keepalive-filtering
IPX circuit number [1]? 5
```

Configuring Multiple Routes

You can configure IPX so that it keeps more than one routing table entry for the same destination network. The benefit of this feature is that if a route goes down, the alternate route is used immediately. The router does not have to wait for a RIP broadcast, which could take from a few seconds to a minute, to learn a new route. The router stores only equal-cost paths in the routing table.

Use the following command to configure the maximum number of routes that will be stored in the routing table for each destination. The range is 1 to 64. The default is 1.

```
IPX config>set maximum routes-per-destination
New maximum number of routes per destination net [1]? 4
```

Use the following command to set the total number of entries kept in the routing table. The range is 1 to 4096. The default is 32. Set the number of entries to at least the same size as the RIP network table. (Configure the size of the RIP network table using the **set maximum networks** command explained in this chapter.)

```
IPX config> set maximum total-route-entries
New route table size [32]? 40
```

Configuring Static Routes

Static routes can be configured per destination network number. Each static route is associated with a circuit and is installed in the routing table when IPX is activated on the circuit. The static route is removed from the routing table when IPX is deactivated on the circuit, the circuit itself is taken down, or any dynamically-learned route to the destination network is learned. Dynamically-learned routes (via RIP) always override static routes. The static route will be reinstalled in the routing table when IPX is reactivated on the circuit, the circuit itself comes back up, or when all RIP routes to the destination network are lost.

Static routes are particularly useful over dial-on-demand circuits where RIP is disabled and routes to destination networks are statically configured on the dial-on-demand circuit.

Static routing may be used on a circuit by itself or in combination with RIP. The only exception to this is when static routing is enabled on an IPXWAN circuit. In this case, static routing is the only routing type negotiated by IPXWAN.

Static routes will be advertised by RIP, subject to split-horizon and applicable filters.

When multiple static routes per destination network are configured, the same rules used to choose RIP routes are used to determine which static routes are installed in the routing table. Multiple static routes to the same destination network will be installed in the routing table if they are of equal cost. Up to the configured routes per destination can be concurrently stored in the routing table.

The following example shows how to configure an IPX static route.

```
IPX Config> disable rip
IPX circuit number [1]? 2

IPX Config> enable route-static

IPX Config> add route-static
IPX net address: (1-fffffffe) [1]? 30
IPX circuit number [1]? 2
Next-hop address, in hex [] ? 400000003000
Ticks: (0-30000) [0]? 4
Hops: (0-14) [0]? 4
```

Configuring Static Services

Static services can be configured per service type or name pair. Each static service is associated with a circuit and is installed in the SAP services table when IPX is activated on the circuit, and a route to the service's network is known (either by static route or RIP advertisement). The static service is removed from the SAP table when IPX is deactivated on the circuit, the circuit itself is taken down, the route to the server's network is lost, or the same service is learned dynamically. As long as

a route to the server's network is known, the static service will be reinstalled in the service table when IPX is reactivated on the circuit, the circuit itself comes back up, or when the SAP-learned service is lost. Dynamically-learned services (using SAP) always override static services.

Static services are particularly useful over dial-on-demand circuits where SAP is disabled and services are statically configured on the dial-on-demand circuit.

Static services may be used on a circuit by itself or in combination with RIP/SAP. The only exception to this is when static routing is enabled on a IPXWAN circuit. In this case, static routing is the only routing type negotiated by IPXWAN.

Static services will be advertised by SAP, subject to split-horizon and applicable filters.

When multiple static services per name or type are configured, the same rules used to choose SAP services are used to determine which static service is installed in the routing table. Note that if there are equal-cost static services configured, the one defined on the same circuit as the current route to the server's network will be installed in the service table.

The following example shows how to configure an IPX static service.

```
IPX Config> disable sap
IPX circuit number [1]? 2

IPX Config> enable sap-static

IPX Config> add sap-static
Sap type: (0-ffff) [4]?
Sap name: []? FILE_SERVER01
IPX circuit number [1]? 2
IPX net address: (1-fffffffe) [1]? 30
IPX node address, in hex: []? 400000202000
IPX socket: (0-ffff) [451]?
Hops: (0-14) [0]? 4
```

Configuring the RIP Default Route

The default route is a special case of a static route. It is used as a last resort as a next hop for unknown destination networks.

The default route is especially useful on dial-on-demand circuits when RIP is disabled. Configuring the default route on the dial-on-demand circuit allows clients to request routes and send packets to destination networks on the other side of the circuit without having to configure a static route for each destination.

RIP Handling

For routers using RIP, the default route is designated by network number FFFFFFFE.

When advertising RIP routes, the default route (like any other static route) will be advertised, after being subjected to the RIP filters and split-horizon.

When responding to a RIP request for an unknown destination network, the router responds to the request only if it has a default route in the routing table.

When forwarding packets, if the route to the destination network is unknown, the forwarder will forward the packet to the next-hop router that is advertising the default route (or the next-hop router indicated by the local static default route definition in the case of static routing).

The following example shows how to configure a RIP default route.

```
IPX Config> enable route-static

IPX Config> add route-static
IPX net address: (1-fffffffe) [1]? fffffffe
IPX circuit number [1]? 2
Next-hop address, in hex: []? 400000003030
Ticks: (0-30000) [0§? 4
Hops: (0-14) [0]? 4
```

Interaction with SAP

Generally, SAP advertisements are accepted only if a route to the server's network is known. If the route to the server's network is not known, but a default route is known, the advertisement is also accepted (after being subjected to the SAP filters).

SAP advertisements that are accepted by virtue of the existence of the default route will be advertised on all IPX circuits other than the one from which the SAP advertisement was accepted (split-horizon). Of course, the advertisement will be subjected to the SAP filters before being advertised. The same rules apply to responses to SAP requests.

Configuring Global IPX Filters (IPX Access Controls)

Global IPX filters are applied to all IPX circuits. They can be used to prevent the router from forwarding packets based on IPX addresses (network/host/socket). You can use global IPX filters to provide security or to stop the forwarding of packets from "noisy" applications beyond the area of interest.

Global IPX filters are based on the originating IPX source address and the ultimate destination IPX address. Intermediate hop addresses are not important.

An IPX address (source or destination) for a global filter consists of an IPX network number, an IPX host number, and a range of IPX socket numbers that are specified in hexadecimal. The network number and host number can be specified as 0, which is a wildcard that matches all network and host numbers, respectively. A range of 0 to FFFF is a wildcard for sockets.

The global filter list is an ordered list of entries. Each global filter entry can be configured as inclusive or exclusive. The router compares packets it receives against the global filter list.

- If a packet matches an inclusive entry, the router forwards the packet.
- If a packet matches an exclusive entry, the router drops the packet.
- If the router reaches the end of the list without matching the packet to an entry, the router drops the packet. (This is equivalent to having a wildcard exclusive entry at the end of the list.)

When creating global filter lists, consider the following things about IPX:

- First, never block the RIP and SAP sockets (X'0453' and X'0452'). RIP and SAP are required to correctly forward IPX packets.

- Remember that the global filter list applies to all circuits. You will have to use source and/or destination network numbers in the global filters to enact directional controls.
- Understand where the services you are trying to protect are located. At the `IPX>` prompt, enter the **slist** command to determine the address of a service.

  Note: All services on a Novell file server (version 3.0 or higher) are on the server's internal network, usually at host 000000000001. Because that internal network number is unique over an entire IPX network, you can protect it by blocking all packets to the internal network socket range 0–FFFF. To block only the file server, use a socket range of 0451–0451.
- When extracting socket numbers from an **slist** to build a global filter list, remember that some services have fixed socket numbers and some have dynamic (temporary) socket numbers. Because sockets in the range 4000–7FFF are dynamic, there is no guarantee that the service will have the same socket number the next time the file server is rebooted. However, socket numbers in the range 8000–FFFF are assigned by Novell, and will generally remain constant.

Note: The global filters and circuit filters are mutually-exclusive. If global SAP filtering is enabled, circuit SAP filters cannot be enabled (and vice versa). If global IPX filtering is enabled (*access-controls*), circuit IPX filters cannot be enabled (and vice versa).

The router examines each IPX frame to see if it matches an entry in the global filter list. It applies the first match, therefore the order of global filters is critical. The router examines IPX packets for the following criteria:

1. Type of global filter (two types):
   a. Inclusive, indicating that if the packet matches the following criteria, forward it
   b. Exclusive, indicating that if the packet matches the following criteria, discard it
2. Destination network - taken directly from the packet's IPX destination network field.
3. Destination host - taken directly from the packet's IPX destination host field.
4. Starting/Ending destination socket - taken directly from the packet's IPX destination socket field (not host field). (The socket number is the location within the protocol that binds the packet to an application service.)
5. Source network - taken directly from the packet's IPX source network field.
6. Source host - taken directly from the packet's IPX source host field.
7. Starting/Ending source socket - taken directly from the packet's IPX source socket field.

The result of the following example would be to forward only those IPX packets from any client on IPX net 1871, destined for the NCP application, on the Novell File Server 0000 C93A 0912, on network 18730. All other traffic would be dropped.

```
IPX config>add access control
Enter type [E]? I
Destination network number (in hex) [ ]? 18730
Destination host number (in hex) [ ]? 0000C93A0912
Starting destination socket number (in hex) [ ]? 0451
Ending destination socket number (in hex) [ ]? 0451
Source network number (in hex) [ ]? 1871
Source host number (in hex) [ ]? 0
Starting source socket number (in hex) [ ]? 4000
Ending source socket number (in hex) [ ]? 7FFF
```

Global SAP Filters

Global SAP filters apply to all circuits. They can be used to prevent service advertising information from being propagated through the router. There are four primary reasons to use global SAP filters:

- You are using servers with small bindery sizes (for example, NetWare Version 2.15 or lower) and must limit the amount of information in the SAP database.
- You do not want to advertise certain services outside the local area, because remote access to them would be inappropriate.
- You want to remove clutter from the SAP table.
- You want to reduce needless SAP advertisements on WAN links, since SAP advertisements can consume a considerable amount of WAN bandwidth.

Note: None of these reasons explicitly mentions security. Global SAP filters cannot protect a service. All that SAP does is provide a name-to-address translation for services. If a potential intruder knows the address of the service, blocking its advertisement via global SAP filters will not protect the service. Only access controls can provide security.

The global SAP filter is based on setting a maximum hop count for a particular service, or group of services. Any matching service advertisement received with the specified hop count (or less) is accepted into the SAP table. Others are ignored. Only those services in the SAP database are re-advertised or used to answer queries.

Note: The router allows you to enter service names in 7-bit ASCII only. Some service names use binary data, in violation of Novell SAP specifications. You will not be able to filter those services by name.

A global SAP filter can apply to all services of a type. Novell assigns 4-digit hexadecimal type numbers for each type of service. Alternately, a global SAP filter can apply to one particular service of a type. This is done by specifying the name of the service.

There can be several servers of the same service type, each with a unique service name. In this case, you can configure multiple global SAP filters with the same service type to filter unique service names, or you can configure a single SAP filter which filters the service type for all service names (wildcard filter).

Creating Global SAP Filters

To configure global SAP filters:

1. Enter **add filter** at the `IPX Config>` prompt. You must specify several key entries that are normally found in the SAP broadcasts:
   a. Number of hops. This entry indicates the hop count allowed for a SAP entry (if higher, discard).
   b. Service type
   c. Service name
2. Enter **set filter on** at the `IPX Config>` prompt to enable the filter.

The following example shows the creation of a global SAP filter against a specific print server.

```
IPX config> add filter
Maximum number of hops allowed [1]? 2
Service type [4]? 0047
Optional service name [ ]? rem-ptr1
IPX config> set filter on
```

This global SAP filter causes the router to ignore SAP advertisements from any print server (service type 0047) named **rem-ptr1** that is more than two hops away. The filter prevents the router from propagating advertisements that match these criteria.

Determining the Service Type for a Global SAP Filter

To determine the SAP type for a filter you want to establish, follow these steps:

1. At the * prompt, enter **talk 5**. Then, at the + prompt, enter **protocol ipx**.

   At the IPX> prompt enter **slist**. Note the entry for the services you want to filter.
2. At the * prompt, enter **talk 6**. Then, at the Config> prompt, enter **protocol ipx**. Add the appropriate global SAP filter and the appropriate hop count for the service you want to filter.
3. After creating the filter, restart the router.
4. If you have successfully filtered a service, it should no longer be listed. Check that the service is no longer listed by entering **slist** at the IPX> prompt.

IPX Circuit Filters - Overview

The IPX routing feature supports four types of circuit-based filters: ROUTER, RIP, SAP, and IPX. One *input* and one *output filter* can be defined per circuit. Filter criteria, referred to as *items*, are assembled into *filter-lists* and are then attached to the input and/or output filters. A filter-list can be attached to more than one filter. This prevents you from having to configure the same filter criteria on multiple circuits.

Note: The global filters and circuit filters are mutually-exclusive. If global SAP filtering is enabled, circuit SAP filters cannot be enabled (and vice versa). If global IPX filtering is enabled (*access-controls*), circuit IPX filters cannot be enabled (and vice versa).

Configuring IPX circuit Filters

To configure IPX circuit Filters:

1. Create a filter-list and give it a name, using the **create list** command.
2. Modify the filter-list using the **update** command and its subcommands to specify the filter criteria and whether this filter-list is inclusive or exclusive.
3. Create a filter on the desired circuit using the **create filter** command, specifying whether it is an input or output filter.
4. Enable IPX circuit filtering using the **enable all** command.
5. Attach filter-lists to the filter using the **attach** command.
6. Set the default action for the filter using the **default** command. The default action will be taken if no match is made on any of the attached filter-lists.

There are also commands to delete a filter on an IPX circuit, disable a filter on an IPX circuit (or all IPX circuits), detach a filter-list from a filter, move the filter-lists within the filter (because the filter-lists are ordered), list a filter, and set the size of the filter cache (for IPX Filtering only).

ROUTER Filtering

The ROUTER Filter operates on the IPX header of all received RIP response packets. Output ROUTER filtering is not supported. ROUTER filtering can be used to group individual IPX networks into several distinct IPX internets by controlling which routers are allowed to exchange routing information.

RIP ROUTER Filters are kept in ordered lists of items by circuit. The items are applied in order to each received RIP response packet. If a match is found, the action specified in the matching filter-list is performed (Exclude = discard packet, Include = receive packet for processing). Because Excluded packets are discarded, the information contained in their network entries is not entered into the RIP routing tables. If no match is found, the specified default filter action is performed.

RIP Filtering

The RIP filter operates on the network entries of RIP response packets. It can be used to control the extent to which routing information about selected networks is disseminated. As an *input* filter, this filter can prevent the *storing* of routing information about selected networks. This prevents ***all*** other networks from learning about the selected networks (at least through this router).

RIP filters (input) are kept in ordered lists of items by circuit. The items are applied in order to each network entry in each received RIP response packet. If a match is found, the action specified in the matching filter-list is performed (Exclude = ignore network entry, Include = process network entry). Because Excluded network entries are ignored, they are not entered into the RIP routing tables. If no match is found, the specified default filter action is performed.

As an *output* filter, this filter can prevent the *advertising* (as opposed to the storing) of routing information about selected networks. It prevents *some* (as opposed to all) networks from learning about the selected networks (at least through this router).

RIP filters (output) are kept in ordered lists of items by circuit. The items are applied in order to each network entry to be transmitted in a RIP response packet. If a match is found, the action specified in the matching filter-list is performed (Exclude = exclude network entry from packet, Include = include network entry in packet). This filter has no effect on the contents of the RIP routing tables. If no match is found, the specified default filter action is performed.

SAP Filtering

The SAP filter operates on the server entries of all SAP response packets. It can be used to control the extent to which information about services is disseminated, and can reduce the amount of SAP traffic on lower speed WANs.

As an *input* filter, this filter can prevent the *storing* of service information about selected servers. This prevents ***all*** other networks from learning about the selected servers (at least through this router).

SAP filters (input) are kept in ordered lists of items by circuit. The items are applied in order to each server entry in each received SAP response packet. If a match is found, the action specified in the matching filter-list is performed (Exclude = ignore server entry, Include = process server entry). Because Excluded server entries are ignored, they are not entered into the SAP services table. If no match is found, the specified default filter action is performed.

As an *output* filter, this filter can prevent the *advertising* (as opposed to the storing) of service information about selected servers. This prevents *some* (as opposed to all) networks from learning about the selected servers (at least through this router).

SAP filters (output) are kept in ordered lists of items by circuit. The items are applied in order to each server entry in each SAP response packet to be transmitted. If a match is found, the action specified in the matching filter-list is performed (Exclude = exclude server entry, Include = include server entry in packet). This filter has no effect on the contents of the SAP services table. If no match is found, the specified default filter action is performed.

IPX Filtering

The IPX Filter operates on the IPX header of IPX packets. It can be used to control the extent to which selected servers and workstations are allowed to communicate with other selected servers and workstations, based on source and destination network, node, and socket fields, as well as protocol type and hop count.

As an *input* filter, a match that indicates that the packet should be discarded prevents the packet from being transmitted on ***all*** circuits.

IPX Filters (input) are kept in ordered lists of items by circuit. The items are applied in order to each received IPX packet. If a match is found, the action specified in the matching filter-list is performed (Exclude = discard packet, Include = receive packet for processing or forwarding). If no match is found, the specified default filter action is performed.

As an *output* filter, the decision whether to forward the packet is made based on the output circuit, and therefore might allow a received packet to be forwarded out on one circuit but not out on some other circuit.

IPX filters (output) are kept in ordered lists of items by circuit. The items are applied in order to each IPX packet to be transmitted. If a match is found, the action specified in the matching filter-list is performed (Exclude = discard packet, Include = transmit packet). If no match is found, the specified default filter action is performed.

Because IPX filters are invoked for each received packet, it is recommended that they be used only where a high degree of specificity is required (that is, where the ROUTER, RIP and SAP filters cannot be used). Generally, the RIP filters deal with internetworking between ***all*** stations on a particular set of networks; the SAP filters control which servers are reachable by workstations throughout the internetwork; the IPX filters deal with internetworking between ***individual*** workstations (or individual applications on individual workstations).

“IPX circuit Circuit-Filter Configuration Commands” on page 406 describes in more detail the commands used to configure IPX circuit Filters.

IPX Performance Tuning

The IPX router implements a dual path for packet forwarding, a fast path and a slow path, to route traffic more efficiently.

The fast path forwards only data packets, while a slower path handles administration packets, such as RIP and SAP packets. Fast path uses an address cache that enables the router to forward a packet quickly.

The slower routing table lookups are performed only during the creation of a cache entry. The cache has an aging mechanism that allows overflows to be dealt with intelligently. You can configure the cache size through the IPX configuration menu.

The IPX fast path cache includes two entries: local and remote. Each entry can handle the requirements of that type of addressing.

The cache commands are used to set a limit on the maximum number of entry types allowed in the cache.

Local Cache

The size of the local cache should equal the total number of clients on each router's local or client network plus a 10% buffer to prevent excessive purge requests. Using the example in Figure 27 on page 383, router 5 (RTR R5) has 9 clients (C) plus the server (S) for a total of 10. Based on this total:

1. Multiply by 10% (10 in our example).
2. Add that total (1) to the client total (for a safety margin).
3. Use the new total (11) for the number of local cache entries.

For example:

```
IPX config>set local-cache size
New IPX local node cache size [32]? 11
```

When all cache entries are in use, the least frequently used entries are purged.

Remote Cache

The size of the remote cache should equal the total number of remote networks used by the router plus a 10% buffer to prevent excessive purge requests. In Figure 27 on page 383, there are 10 IPX networks that RTR R5 can read via IPX network 5. Therefore, RTR/R5 has a total of 10 clients. Based on this total:

1. Multiply by 10% (10 in our example).
2. Add that total (1) to the remote network total (10) for a safety margin.
3. Use the new total (11) for the number of remote cache entries.

For example:

```
IPX config>set remote-cache size
New IPX remote network cache size [32]? 11
```

You can view the cache entries using the IPX monitoring **sizes** command.

```
IPX>sizes
Current IPX cache size:
Remote network cache size (max entries): 45
0 entries now in use
Local node cache size (max entries): 86
0 entries now in use
```

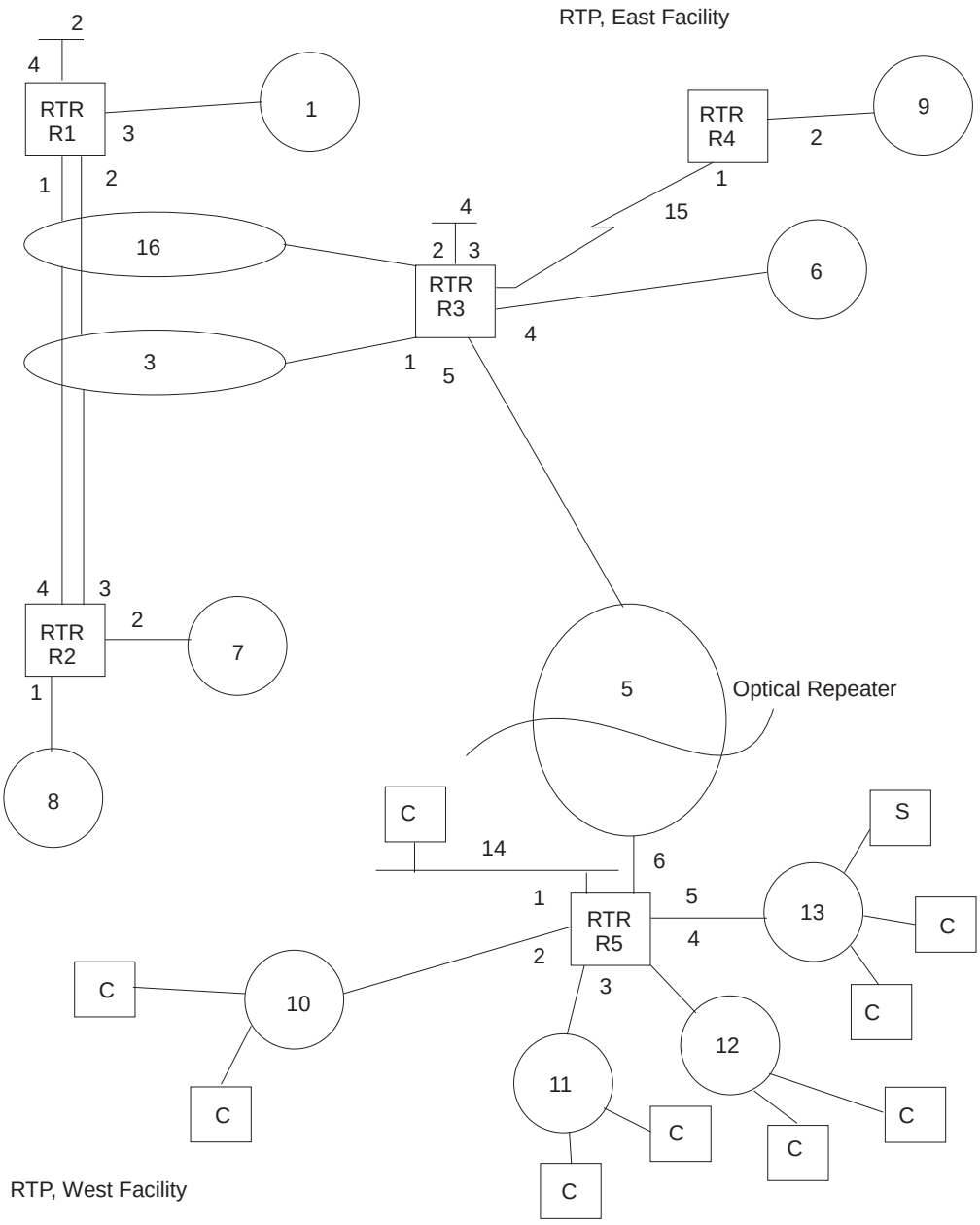

Figure 27. Sample IPX Network

Split-Horizon Routing

Split-horizon is a method of routing that avoids broadcasting RIP and SAP updates to the router from which they were learned.

Generally, split-horizon should be enabled on every circuit to prevent packets from counting to infinity and to avoid unnecessary RIP and SAP advertisements. However, there are some cases, such as partially-meshed frame-relay, ATM, and X.25 configurations, where it may be necessary to disable split-horizon.

A Partially-meshed RFC 1483-Supported IPX Routing configuration is another case where it may be necessary to disable split-horizon.

In a partially-meshed frame-relay network, as shown in Figure 28, the routers at the branches cannot communicate with each other unless the router at headquarters broadcasts all routing information to all other routers. In this case, split-horizon should be disabled on the frame-relay circuit at headquarters, and enabled at each of the branches to keep them from generating unnecessary traffic.

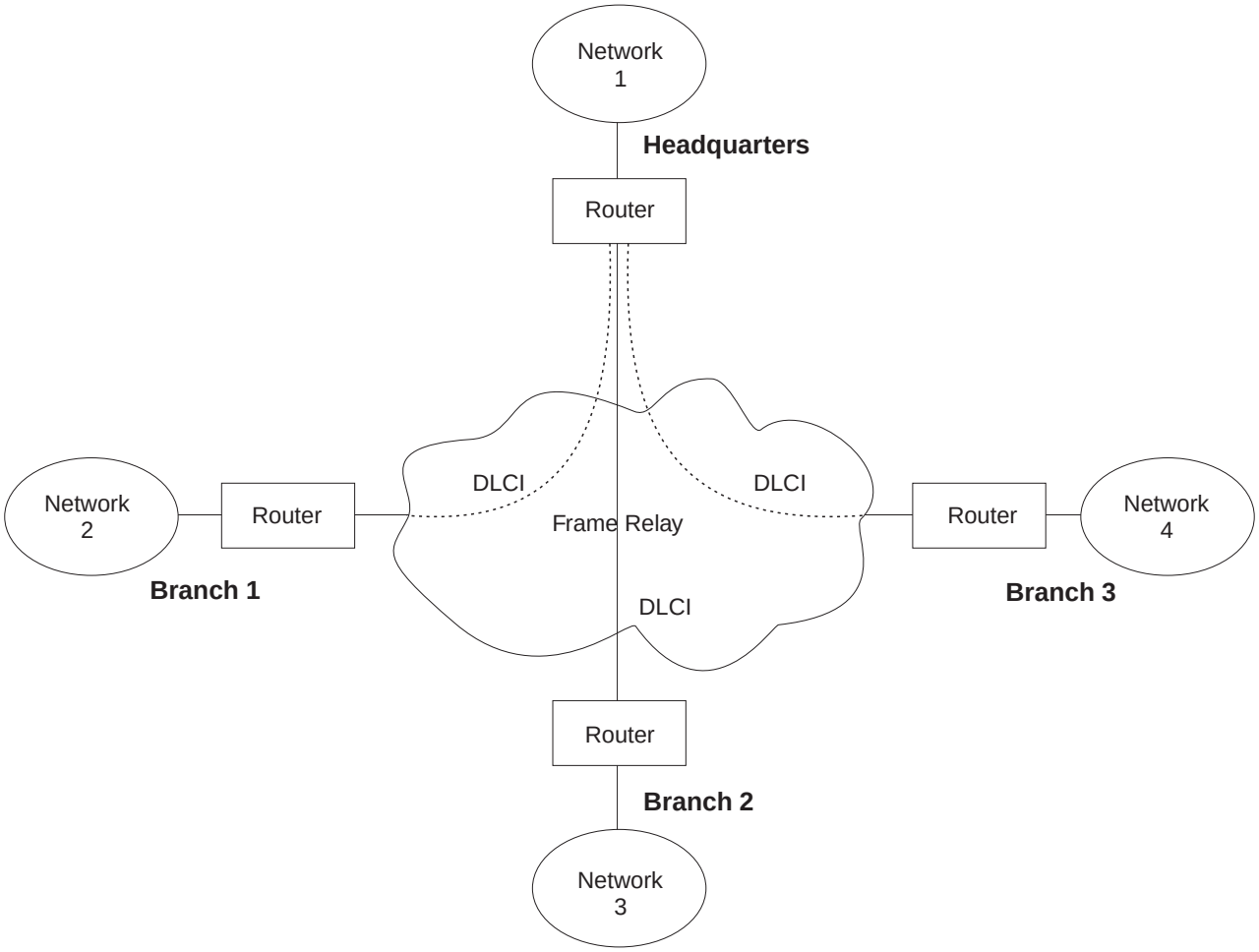

Figure 28. Partially Meshed Frame-Relay Network

If you do need to change the split-horizon setting, use the **set split-horizon** command as follows:

```
IPX Config>set split-horizon enabled
Which circuit [1]? 2

IPX Config>set split-horizon disabled
Which circuit [1]? 2

IPX Config>set split-horizon heuristic
Which circuit [1]? 2
```

Chapter 32. Configuring and Monitoring IPX

This chapter describes how to configure the IPX protocol and use the IPX monitoring commands. It includes the following sections:

Accessing the IPX Configuration Environment

To access the IPX configuration environment, enter the following command at the `Config>` prompt:

```
Config> protocol IPX
IPX Protocol user configuration
IPX Config>
```

IPX Configuration Commands

This section discusses the IPX configuration commands. Table 60 lists the IPX configuration commands. These commands specify the network parameters for routers transmitting IPX packets. These commands are entered at the `IPX config>` prompt. To activate the configuration changes, restart the router.

Table 60. IPX Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds an IPX broadcast or IPXWAN point-to-point circuit, adds global IPX filters (access controls), global SAP filters, static routes or services.
Delete	Deletes an IPX broadcast or IPXWAN point-to-point circuit, deletes global IPX filters (access controls), global SAP filters, static routes, or services.
Disable/Enable	Disables or enables IPX globally or on specific IPX circuits, globally disables or enables the use of IPX static routes or services. Disables or enables Keepalive filtering, RIP-SAP broadcast pacing, SAP reply to get nearest server, NetBIOS broadcasts, and disables or enables RIP or SAP on specific circuits.
Filter-lists	Accesses the IPX circuit-filter configuration. This environment is where the IPX circuit-based ROUTER, RIP, SAP, and IPX filters are configured.
Frame	Specifies the data link format for Ethernet and Ethernet LAN Emulation Clients.
List	Displays the current IPX configuration.
Move	Reorders the global IPX filter items (access control), or moves an IPX circuit from one interface to another.

© Copyright IBM Corp. 1996, 1999

Table 60. IPX Configuration Commands Summary (continued)

Command	Function
Set	Sets the host number, IPXWAN router name and node ID, IPXWAN routing type, connection timeout and retry timer, IPX network numbers, maximum RIP and SAP table sizes, local and remote cache sizes, global IPX filter (access controls) and global SAP filter states, cache sizes, RIP and SAP update intervals, RIP circuit cost (RIP ticks), Keepalive filtering table size, and split-horizon usage.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **add** command to add a global IPX filter (access controls), an IPX broadcast circuit, a global SAP filter, an IPX point-to-point circuit, or a static route or service to your IPX configuration.

Syntax:

add access-control . . .
broadcast-circuit . . .
filter . . .
route-static . . .
sap-static . . .

access-control *type dest-net dest-host dest-socket-range src-net src-host src-socket-range*

Determines whether to pass a packet at the IPX level. IPX access controls provide a global access control function at the IPX packet level for the IPX protocol. The access control list is an ordered set of entries that the router uses to filter packets. Each entry can be either Inclusive or Exclusive. Each entry has source and destination network numbers, host addresses, and socket ranges.

When a packet is received from a network for the IPX protocol, and access control is enabled, it is checked against the access control list. It is compared with the net/address/socket pairs in the list until there is a match. If there is a match and the entry is of the Inclusive type, reception of the packet (and potential forwarding) proceeds. If the matching entry is of the Exclusive type, the packet is dropped. If there is no match, the packet is also dropped.

After you create an access-control list with the **add access-control** command, enable the entries with the **set access-control on** command. Use the **move** command to change the order of the access-control list.

Note: Access controls apply to all received packets. If you do not enable reception of RIP (socket 453 hexadecimal) or SAP (socket 452 hexadecimal) packets, the IPX forwarder will be nonfunctional.

```
add access I 0 0 453 453 0 0 0 FFFF
add access I 0 0 452 452 0 0 0 FFFF

Enter type [E] i
Destination network number (in hex) [0]? 0
Destination host (in hex) [ ]? 0
Starting destination socket number in hex [0]? 452
Ending destination socket number in hex [0]? 453
Source network number (in hex) [0]? 0
Source host number (in hex) [ ]? 0
Starting source socket number in hex [0]? 0
Ending source socket number in hex [452]? FFFF
```

Type
Identifies whether packets are sent or dropped for a specific address or set of addresses. Enter I for include. This causes the router to receive the packet and to forward it if it matches criteria in the remaining arguments. Enter E for exclude. This causes the router to discard the packets.

Dest-net
Network number of the destination. Enter the network number in hexadecimal.

Valid Values: X'00000000' to X'FFFFFFFF'

Zero (0) specifies all networks.

Default Value: 0

Dest-host
Host number on the destination network. Enter the host number in hexadecimal.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Zero (0) specifies all hosts on the network.

Default Value: None

Dest-socket-range
Two numbers that specify an inclusive range of destination sockets. The destination socket value is used for filtering IPX packets.

Valid Values: X'0000' to X'FFFF'

Default Value: 0

Src-net
Network number of the source. Enter the network number in hexadecimal.

This parameter defines the network number of the source IPX network whose packets are filtered by this router.

If you choose to filter on *only* the source network value, the filter applies to all source sockets, source networks, packet types, and number of hops.

Valid Values: X'00000000' to X'FFFFFFFF'

Zero (0) specifies all networks.

Default Value: 0

Src-host
Host number on the source network. Enter the host number in hexadecimal.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Zero (0) specifies all hosts on the network.

Default Value: None

Src-socket-range
Two numbers that specify an inclusive range of source sockets.

Valid Values: X'0000' to X'FFFF'

Default Value: 0

Note: It is not necessary to use access controls and SAP filters for IPX to work in a NetWare environment. Use them only if necessary.

Example: `add access-control E 201 1 451 451 329 0 0 FFFF`

This access control prevents all nodes on network 329 from accessing the file server with internal network number 201.

broadcast-circuit *interface# ipx-circuit# network#*
Adds an IPX broadcast circuit.

interface#
Specifies the network interface on which the IPX circuit number is configured.

Valid Values: valid network interface number

Default: 0

ipx-circuit#
Specifies the IPX circuit number. This number must be unique among all configured IPX circuits in the router and is used to reference IPX circuits in many of the configuration commands.

Valid Values: 1- 65535

Default: next available IPX circuit number

network#
Specifies the IPX network number to be used on the IPX circuit. IPX network number 0 is valid only on IPXWAN unnumbered RIP or static routing circuits. IPX network number FFFFFFFF is not a valid IPX network number. IPX network number FFFFFFFE is reserved for the IPX Default Route and may not be used as an IPX network number.

Valid Values: 1 - FFFFFFFD

Default: 1

Example:

```
add broadcast-circuit
Which interface [0]?
IPX circuit number [1]?
IPX network number in hex
(0 is allowed only on IPXWAN unnumbered circuits) [1]? 400
```

filter *hops service-type service-name*
Prevents NetWare bindery overflows for users on large networks by enabling you to determine the number of hops reasonable for a given

service. IPX SAP filters allow the protocol to be configured to ignore certain entries in SAP advertisements. This is done to limit the size of the SAP database. This could be necessary due to size limitations in older versions of NetWare file servers. This could also be necessary to limit the amount of SAP data sent across WAN links.

The SAP filters are a global ordered list of filter entries. Each filter entry has a maximum hop count, a service type, and an optional service name. When a SAP response packet is received, each SAP entry is compared with the filter list. If the SAP entry matches an entry in the filter list and is greater than the specified hops, it is ignored and not entered into the local SAP database. If the SAP entry matches an entry in the filter list, and is less than or equal to the specified hops, it is accepted and entered into the local SAP database. If there is no match, the SAP entry is accepted. The arguments for this command are as follows:

Hops
Maximum number of hops permitted for the service.

Valid Values: An integer in the range of 0 to 16.

Default Value: 1

Service-type
Numeric service class.

Valid Values: A hexadecimal value in the range of X'0000' to X'FFFF'.

Use a value of X'0000' to filter all service types.

Default Value: 4

You can see a list of service types by entering the **slist** command at the `IPX>` prompt.

Service-name
Identifies the name of the server. In general, this field is not entered.

Valid Values: A string of 1 to 47 ASCII characters (X'20' through X'7E').

Default Value: none

Example: `add filter 2 039B NOTES-CHICAGO`

This example ignores all SAP advertisements for the Lotus Notes server “NOTES-CHICAGO” at more than 2 hops.

route-static *dest-net ipx-circuit# nextHop ticks hops*
Adds a static route.

dest-net
Specifies the destination IPX network number.

Valid Values: X'1' to X'FFFFFFFE'

Default Value : 1

ipx-circuit#
Specifies an existing IPX circuit on which the static route should be configured.

Valid Values: existing IPX circuit number

Default Value: 1

nextHop
Specifies the IPX host number of the next-hop router through which the destination network can be reached.

Valid Values: X'1' to X'FFFFFFFFFFFE'

Default Value: none

ticks
Indicates the number of ticks between the destination network and this router. The number of ticks represents the amount of time it takes to transmit a 576-byte IPX packet from this router to the destination network. Each tick is 55 milliseconds.

Valid Values: 0 to 30000

Default Value: 0

hops
Indicates the number of hops between the destination network and this router.

Valid Values: 0 to 14

Default Value: 0

Example:

```
add route-static
IPX net address: (1-fffffffe) [1]? 30
IPX circuit number [1]? 2
IPX node address (in hex) []? 020000002030
Ticks: (0-3000) [0]? 4
Hops: (0-14) [0]? 4
```

sap-static *serviceType serviceName ipx-circuit# serverNet serverNode serverSocket hops*
Adds a static SAP service.

serviceType
Specifies the hexadecimal service class of the service.

Valid Values: X'0' to X'FFFF'

Default Value: 4

serviceName
Specifies the ASCII name of the service.

Valid Values: up to 47 of the following ASCII characters: 'A'–'Z', 'a'–'z', '0'–'9', '_', '-', '@'.

Default Value: None

ipx-circuit#
Specifies an existing IPX circuit on which the SAP static service should be configured.

Valid Values: existing IPX circuit number

Default Value: 1

serverNet
Specifies the internal IPX network number or home IPX network number of the server.

Valid Values: X'1' to X'FFFFFFFE'

Default Value: 1

serverNode
Specifies the IPX node of the server.

Valid Values: X'1' to X'FFFFFFFFFFFE'

Default Value: None

serverSocket
Specifies the socket number of the server.

Valid Values: X'0' to X'FFFF'

Default Value: 451

hops
Indicates the number of hops between the server and this router.

Valid Values: 0 to 14

Default Value: 0

Example:

```
add sap-static
Sap type: (0-ffff) [4]? 4
IPX circuit number [1]? 2
IPX net address: (1-fffffffe) [1]? 40
IPX node address, in hex:  []? 000000000001
IPX socket:  (0-ffff) [451]?
Hops: (0-14) [0] 4
```

Delete

Use the **delete** command to delete an IPX broadcast or IPXWAN point-to-point circuit, a global IPX filter (access control), a global SAP filter, a static route or a static service.

Syntax:

delete access-control . . .
circuit . . .
filter . . .
route-static . . .
sap-static . . .

access-control *line#*

Deletes the access control that matches the line number you enter. Enter the **list** command to display the current line numbers.

Example: `delete access-control 2`

circuit *ipx-circuit#*

Deletes the IPX broadcast or IPXWAN point-to-point circuit. It will also delete all of the static routes, static services and circuit filters that are associated with the specified *ipx-circuit#*.

Example: `delete circuit`

```
IPX circuit number [1]? 2
You are about to delete IPX broadcast circuit 2 on interface 4.
All associated static routes, static services and circuit filters
will be deleted as well.  Are you sure? [Yes]: yes
```

filter *hops service-type service-name*

Deletes the specified SAP filter. You must type the SAP filter exactly as it appears when you run the list command. The arguments are as follows:

Hops

Maximum number of hops permitted for the service.

Valid Values: 0 to 16

Default Value: 16

Service-type

Numeric service class. Enter a 2-byte hexadecimal number.

Valid Values: X'0000' to X'FFFF'

Default Value: None

Service-name

If the entry you are deleting has a name, specify the name.

Valid Values: A string of 1 to 47 ASCII characters (X'20' through X'7E').

Default Value: None

Example: `delete filter 2 039B NOTES-CHICAGO`

route-static *dest-net ipx-circuit# nextHop*

Deletes a static route.

dest-net

Specifies the destination IPX network number.

Valid Values: X'1' to X'FFFFFFFE'

Default Value: 1

ipx-circuit#

Specifies the IPX circuit on which the static route is configured.

Valid Values: existing IPX circuit number

Default Value: 1

nextHop
Specifies the IPX host number of the next-hop router through which the destination network can be reached.

Valid Values: X'1' to X'FFFFFFFFFFFE'

Default Value: none

Example:

```
delete route-static
IPX net address: (1-fffffffe) [1]? 30
IPX circuit number [1]? 2
IPX node address (in hex) []? 020000002030
```

sap-static *serviceType serviceName ipx-circuit#*
Deletes a static SAP service.

serviceType
Specifies the hexadecimal service class of the service.

Valid Values: X'0' to X'FFFF'

Default Value: 4

serviceName
Specifies the ASCII name of the service.

Valid Values: up to 47 of the following ASCII characters: 'A'–'Z', 'a'–'z', '0'–'9', '_', '-', '@'.

Default Value: None

ipx-circuit#
Specifies the IPX circuit on which the SAP static service is configured.

Valid Values: existing ipx-circuit number

Default Value: 1

Example:

```
delete sap-static
Sap type: (0-ffff) [4]?
Sap name: (0-ffff) []? filesrv1
IPX circuit number [1]? 2
```

Disable

Use the **disable** command to disable globally or on specific IPX circuits, globally disables the use of IPX static routes and services. Also, use the **disable** command to disable replies to SAP to get-nearest-server , RIP-SAP Broadcast Pacing, RIP, or SAP on specific circuits.

Syntax:

disable circuit . . .
ipx
keepalive-filtering . . .
nebios-broadcast . . .
reply-to-get-nearest-server . . .
rip . . .
rip-sap-pacing . . .

route-static . . .
sap . . .
sap-static . . .

circuit *ipx-circuit#*
Disables the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit.*

Example: `disable circuit`

```
IPX circuit number [1]? 2
```

ipx Globally disables the IPX protocol.

Example: `disable ipx`

keepalive-filtering *ipx-circuit#*
Disables Keepalive-filtering on the IPX broadcast circuit or IPXWAN point-to-point circuits specified by *ipx-circuit#.*

Example: `disable keepalive-filtering`

```
IPX circuit number [1]? 2
```

netbios-broadcast *ipx-circuit#*
Disables receiving and sending Novell NetBIOS broadcasts (packet type 20) on the IPX circuit specified by *ipx-circuit#*. The default is value is enabled. Receiving and sending Novell NetBIOS broadcasts is automatically disabled on IPXWAN static routing circuits, even if it is enabled in the configuration.

Example: `disable netbios-broadcast`

```
IPX circuit number [1]? 2
```

reply-to-get-nearest-server *ipx-circuit#*
Prevents the router from responding to SAP get-nearest-server requests on the IPX broadcast circuit or IPXWAN point-to-point circuit specified by *ipx-circuit#.*

Note: Disabling this feature should be done with great caution. This command should be used only when there are multiple routers (or servers) on an IPX network and it is known that the “best” server is not behind this router.

Example: `disable reply-to-get-nearest`

```
IPX circuit number [1]? 2
```

rip *ipx-circuit#*
Disables RIP on the IPX broadcast circuit or IPXWAN point-to-point circuit specified by *ipx-circuit#*. The default is for RIP to be enabled on all circuits. RIP will automatically be disabled on circuits using IPXWAN Static Routing, even if it is configured as enabled.

Example: `disable rip 1`

rip-sap-pacing *ipx-circuit#*
Prevents RIP/SAP Broadcast Pacing on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#*. When pacing is disabled, RIP and SAP periodic broadcasts are transmitted on the circuit with a 55 msec interpacket gap (the default setting). Enable pacing only on circuits where RIP and SAP broadcasts might cause congestion (for example, you can enable pacing on frame-relay or X.25 circuits with many virtual circuits).

Example: `disable rip-sap-pacing`

```
IPX circuit number [1]? 2
```

route-static
Globally disables the use of static routes.

Example: `disable route-static`

sap *ipx-circuit#*
Disables SAP on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit*. The default is for SAP to be enabled on all circuits. SAP will automatically be disabled on RLAN circuits and on IPXWAN Static Routing, even if SAP is configured as enabled.

Example: `disable sap`

```
IPX circuit number [1]? 2
```

sap-static
Globally disables the use of static services.

Example: `disable sap-static`

Enable

Use the **enable** command to enable IPX globally or on specific circuits. The enable command can also be used to globally enable the use of IPX static routes or services, enables keepalive filtering, RIPS-SAP broadcast pacing, SAP reply to get-nearest-server, RIP or SAP on specific circuits.

Syntax:

enable
- circuit . . .
- ipx
- keepalive-filtering . . .
- nebios-broadcast . . .
- reply-to-get-nearest-server . . .
- rip . . .
- rip-sap-pacing . . .
- route-static . . .
- sap . . .
- sap-static . . .

circuit *ipx-circuit# network#*
Enables the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#* and specifies the IPX network number for the IPX circuit. The IPX circuit will be enabled if a valid IPX network number is configured.

Example: enable circuit

```
IPX circuit number [1]?
IPX network number in hex
(0 is allowed only on IPXWAN unnumbered circuits) [1]?
```

ipx-circuit#
Specifies the IPX broadcast circuit to be enabled.

Valid Values: any valid ipx-circuit number

Default Value: 0

network#
Specifies the IPX network to be used on the circuit. IPX network number 0 is valid only on IPXWAN unnumbered RIP or static routing circuits. IPX network number FFFFFFFF is not a valid IPX network

number. IPX network number FFFFFFFE is reserved for the IPX Default Route and may not be used as an IPX network number.

Valid Values: X'0' to X'FFFFFFFD'

Default Value: 1

Example:

ipx Globally enables the IPX protocol.

Example: `enable ipx`

keepalive-filtering *ipx-circuit#*
Enables Keepalive filtering on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#.*

Example: `enable keepalive-filtering`

```
IPX circuit number [1]? 2
```

netbios-broadcast *ipx-circuit#*
Enables receiving and sending Novell NetBIOS broadcasts (packet type 20) on the IPX circuit specified by *ipx-circuit#*. The default value is enabled. Receiving and sending Novell NetBIOS broadcast is automatically disabled on IPXWAN static routing circuits, even if enabled in the configuration.

Example: `enable netbios-broadcast`

```
IPX circuit number [1]? 2
```

reply-to-get-nearest-server *ipx-circuit#*
Enables the router to respond to SAP get-nearest-server requests on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#*.

Example: `enable reply-to-get-nearest`

```
IPX circuit number [1]? 2
```

rip *ipx-circuit#*
Enables RIP on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#*. The default is for RIP to be enabled on all IPX circuits. RIP is automatically disabled on RLAN circuits and on IPXWAN static routing circuits, even if RIP is enabled in the configuration.

Example: `enable rip`

```
IPX circuit number [1]? 2
```

rip-sap-pacing *ipx-circuit#*
Enables RIP/SAP Broadcast Pacing on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#*.

Note: The router calculates an interpacket gap that guarantees that broadcast completion within the configured RIP and SAP update intervals. Configuring these intervals to a larger value may be necessary for the router to calculate a sufficiently large interpacket gap.

Pacing should be enabled only on circuits where RIP and SAP broadcasts might cause congestion (for example, on frame-relay or X.25 circuits with many virtual circuits).

Example: `enable rip-sap-pacing`

```
IPX circuit number [1]? 2
```

route-static
Globally enables the use of static routes.

Example: `enable route-static`

sap *ipx-circuit#*
Enables SAP on the IPX broadcast or IPXWAN point-to-point circuit specified by *ipx-circuit#*.

Example: `enable sap`

sap-static
Globally enables the use of static services.

Example: `enable sap-static`

Filter-lists

Use the **filter-lists** command to access the `IPX filter-type-List Config>` prompt. Valid filter list types are router, rip, sap, and ipx.

For information about the commands available at the `IPX filter-type.-List Config>` prompt, see "IPX circuit Circuit-Filter Configuration Commands" on page 406.

Syntax:

filter-lists	router-lists
	rip-lists
	sap-lists
	ipx-lists

Example: `filter-lists router-lists`

Frame

Use the **frame** command to specify the packet format for IPX circuits. (Encapsulation can also be set using the CONFIG **network** command.)

Note: When there are incorrect or invalid configuration records, the default frame values are used.

Syntax:

frame	ethernet_II . . .
	ethernet_8022 . . .
	ethernet_8023 . . .
	ethernet_SNAP . . .

ethernet_II *ipx-circuit#*
Sets the frame type to ethernet_II on the IPX broadcast circuit specified by *ipx-circuit#*. The ethernet_II encapsulation uses ethernet version 2.0 with protocol type 8137. This is the NetWare 4.0 and greater default.

Example: `frame ethernet_II`

```
IPX circuit number [1]?
```

ethernet_8022 *ipx-circuit#*
Sets the frame type to ethernet_8022 on the IPX broadcast circuit specified by *ipx-circuit#*. The ethernet_8022 encapsulation uses LLC encapsulation with SAP E0.

Example: `frame ethernet_8022`

```
IPX circuit number [1]?
```

ethernet_8023 *ipx-circuit#*

Sets the frame type to ethernet_8023 on the IPX broadcast circuit specified by *ipx-circuit#*. The ethernet_8023 encapsulation uses ethernet 802.3 encapsulation with no LLC header. This is the pre-NetWare 4.0 default. It is also the router default.

Example: `frame ethernet_8023`

```
IPX circuit number [1]?
```

ethernet_SNAP *ipx-circuit#*

Sets the frame type to ethernet_SNAP on the IPX broadcast circuit specified by *ipx-circuit#*. The ethernet_SNAP encapsulation uses SNAP encapsulation with a PID of 0000008137.

Example: `frame ethernet_SNAP`

```
IPX circuit number [1]?
```

List

Use the **list** command to display the current IPX configuration.

Syntax:

list access-controls
all
circuit
filters
route-static
sap-static
summary

access-controls

Lists the global IPX filters (access-controls). This command displays the information that is displayed in the “Access Control Configuration” section of the **list all**.

all Lists the entire IPX configuration.

Example:

```
 list all

IPX Globals
------------
IPX Globally                Enabled
Host Number (serial line)   020000003024
Maximum Services                  32
Maximum Networks                  32
Maximum Routes                    32
Maximum Routes per Destination    1
Maximum Local Cache entries       64
Maximum Remote Cache entries      64
Keepalive-Filtering Table Size    32




SAP Configuration:
------------------
                                        Update    Split      Broadcast  Get Nearest
Circ   Ifc    NetNum    SAP       Interval  Horizon    Pacing     Reply
1      0      400       Enabled   1         Enabled    Disabled   Enabled
2      1      411       Enabled   1         Enabled    Disabled   Enabled
3      2      412       Enabled   1         Enabled    Disabled   Enabled
```

```
Static Route Configuration:
---------------------------
Static Routes:  Enabled
Dest Net  Hops   Ticks  Next Hop      Circ   Ifc
ABC       3      4      020000003044  3      2

Static Services Configuration:
------------------------------
Static Services:  Enabled
Type Service Name                  Srv Net  Host         Sock Hops Circ   Ifc
4    FILESRV01                     ABC      000000000001 451  3    3      2

SAP Filter Configuration:
-------------------------
IPX SAP Filters:  Enabled
Index Max Hops  Type  Service Name
1     5         4     FILESRV02

Access Control Configuration:
-----------------------------
IPX Access Controls:  Enabled
#   T Dest Net Host         Sock Sock Src Net  Host         Sock Sock
1   E 2        000000000000 0    FFFF 3        000000000000 0    FFFF
2   I 0        000000000000 452  453  0        000000000000 0    FFFF
```

circuit *ipx-circuit#*
Lists the IPX broadcast circuit specified by *ipx-circuit#.* This command displays the information shown in the “IPX Configuration,” “RIP Configuration,” “SAP Configuration,” and “IPXWAN Configuration” sections of the **list all** command example.

filters Lists the global SAP filters. This command displays the information shown in the “SAP Filter Configuration” section of the **list all** command example.

route-static
Lists the static routes. This command displays the information shown in the “Static Route Configuration” section of the **list all** command example.

sap-static
Lists the static services. This command displays the information shown in the “Static Services Configuration” section of the **list all** command example.

summary
Lists a summary of the IPX, RIP, SAP, IPXWAN, and Keepalive filtering configuration for all circuits on which IPX is enabled. This command displays the information shown in the “IPX Globals,” “IPX Configuration,” “RIP Configuration,” “SAP Configuration,” and “IPXWAN Configuration” sections of the **list all** command example.

IPX Globals
The following global information is displayed:
- Whether IPX is globally enabled or disabled
- IPX host number
- Maximum services
- Maximum networks
- Maximum routes
- Maximum routes per destination
- Maximum local cache entries
- Maximum remote cache entries
- Keepalive-filtering table size

IPX Configuration
The following is displayed for each circuit on which IPX is enabled:
- IPX circuit number
- Network interface number
- IPX network number (Netnum)
- IPX is enabled/disabled on circuit

- NetBIOS Broadcast
- Keepalive filtering
- Encapsulation

RIP Configuration

The following information is displayed for each circuit on which IPX is enabled:

- IPX circuit number
- Network interface number
- IPX network number (Netnum)
- Whether RIP is enabled or disabled
- RIP update interval timer
- Whether split-horizon is enabled or disabled
- Whether RIP broadcast pacing is enabled or disabled

SAP Configuration

The following information is displayed for each circuit on which IPX is enabled:

- IPX circuit number
- Network interface number
- IPX network number (Netnum)
- Whether SAP is enabled or disabled
- SAP update interval timer
- Whether split-horizon is enabled or disabled
- Whether SAP broadcast pacing is enabled or disabled
- Whether reply to SAP get-nearest-server request is enabled.

Static Routes Configuration

Displays whether static routes are globally enabled or disabled. In addition, the following is displayed for each configured static route.

- IPX destination network number
- Hops
- Ticks
- Next hop node address
- IPX circuit number
- Network interface number

Static Services Configuration

Displays whether static services are globally enabled or disabled. In addition, the following is displayed for each configured static service:

- Service type
- Service name
- IPX network number of service
- IPX node address of Service (Host)
- Socket
- Hops
- IPX circuit number
- Network interface number

SAP Filter Configuration

Displays whether the global SAP filters are enabled or disabled. In addition, the following information is displayed for each configured global SAP filter:

- Index
- Max hops
- Service type
- Service name

Access Control Configuration

Displays whether the global IPX filters (access controls) are enabled or

disabled. In addition, the following information is displayed for each configured global IPX filter (access control):

- Access control index (#)
- Filter type (include or exclude)
- Destination IPX network number
- Destination IPX node number (Host)
- Destination IPX socket range
- Source IPX network number
- Source IPX node number (Host)
- Source IPX socket range

Move

Use the **move** command to reorder the global IPX filter items (access control), or move an IPX circuit from one interface to another.

Syntax:

move access-control *srcLine# dstLine#*

circuit *ipx-circuit# interface# [ ]FR-circ#]*

access-control *srcLine# dstLine#*

srcLine#
Specifies the line number of the access control you want to move.

dstLine#
Specifies the line number of the access control after which the scrLine should be moved.

After the line is access control is moved, the lines are renumbered.

Example:

```
move access-control
Enter index of control to move [1]? 1
Move record AFTER record number [0]? 2
About to move:
#   T Dest Net Host         Sock Sock Src Net  Host         Sock Sock
1   E 2        000000000000 0    FFFF 3        000000000000 0    FFFF
to be after:
2   I 0        000000000000 452  453  0        000000000000 0    FFFF
Are you sure this is what you want to do? [Yes]: yes
```

circuit *ipx-circuit# interface# [FR-circ#]*
Moves an IPX circuit from one network interface to another. This command also moves all of the static routes, static services, and IPX circuit filters associated with the given *ipx-circuit#* to the same *interface#*.

ipx-circuit#
Specifies the IPX circuit that is to be moved.

Valid Values: an existing IPX circuit number

Default Value: 1

interface#
Specifies the network interface that the IPX circuit is moving to.

Valid Values: an existing network interface number.

Default Value: 0

Set

Use the **set** command to configure the host number, connection timeout and retry timer, IPX network numbers, maximum RIP and SAP table sizes, local and remote cache sizes, global IPX filter (access control) and global SAP filter states, RIP and SAP update intervals, Keepalive filter table size and split-horizon usage.

Syntax:

set access-control . . .
filter . . .
host-number . . .
keepalive-table-size . . .
local-cache size . . .
maximum routes-per-destination . . .
maximum networks . . .
maximum services . . .
maximum total-route-entries . . .
name . . .
net-number . . .
node-id . . .
remote-cache size . . .
rip-update-interval . . .
sap-update-interval . . .
split-horizon . . .

access-control *on* **or** *off*

Turns the global IPX filters (access controls) on or off. Enter **on** or **off**.

Example: `set access-control on`

filter *on* **or** *off*

Turns the global SAP filters on or off. Enter **on** or **off**.

Example: `set filter on`

host-number *host#*

Specifies the host number used for serial circuits running IPX. Each IPX router operating over serial circuits must have a unique host number. This is required because serial circuits do not have hardware node addresses from which to build a host number.

Valid Values: An 12-digit hexadecimal number in the range of X'000000000001' to X'FEFFFFFFFFFF'.

Default Value: none

This number must be unique on each router.

Example: `set host-number 0000000000F4`

Note: IPXWAN requires a router node ID and name to be configured. Use the **set node-ID** and **set name** commands to configure these parameters.

local-cache size *size*

Specifies the size of the local cache routing table.

The size of the local cache should equal the total number of clients on each router's local or client network plus a 10% buffer to prevent excessive purge requests.

Valid Values: The range is 1 to 10000.

Default Value: 64. For more information, see “Local Cache” on page 382 and “Remote Cache” on page 382.

Example: `set local-cache size`

```
New IPX local node cache size [64]? 80
```

maximum routes-per-destination *routes*

Specifies the maximum number of routes per destination network to store in the IPX RIP routes table.

Valid Values: An integer in the range of 1 to 64.

Default Value: 1. For additional information on multiple routes, see “Configuring Multiple Routes” on page 373.

Example: `set maximum routes-per-destination 8`

maximum networks *size*

Specifies the size of the IPX RIP network table. This reflects the number of networks in the internet on which IPX operates.

Valid Values: 1 to 2048

Router memory constraints can prevent the maximum table size from being used.

Default Value: 32 This value cannot be larger than the maximum total-route-entries *size*.

Example: `set maximum networks 30`

maximum services *size*

Specifies the size of the IPX SAP service table. This reflects the number of SAP services in the internetwork on which IPX operates.

Valid Values: 1 to 2048

Router memory constraints can prevent the maximum table size from being used.

Default Value: 32

Example: `set maximum services 30`

maximum total-route-entries *size*

Specifies the size of the IPX RIP routes table. This reflects the total number of routes, including alternate routes, in the internetwork on which IPX operates.

Valid Values: 1 to 4096

Default Value: 32

This value must be at least as large as the *maximum networks size*. For additional information of multiple routes, see “Configuring Multiple Routes” on page 373.

Example: `set maximum total-route-entries 40`

name *router_name*

Lets you assign a symbolic name to the router. IPXWAN requires a router to have a node id and name.

Valid Values: A variable length string of 1 to 47 characters.

The router_name can contain the characters A through Z, 0 through 9, underscore (_), hyphen (-), and ″at″ sign (@).

Default Value: none.

Example:
```
set name newyork_accounting
```

net-number *ipx-circuit# network#*

Specifies the IPX network number fro the IPX broadcast or IPXWAN point-to-point circuit.

ipx-circuit#

Specifies an existing IPX broadcast or IPXWAN point-to-point circuit.

Valid Values: an existing circuit number

Default Value: 1

network#

Specifies the IPX network number to be used on the IPX circuit. IPX network number 0 is valid only on IPXWAN unnumbered RIP or static routing circuits. IPX network number FFFFFFFF is not a valid IPX network number. IPX network number FFFFFFFE is reserved for the IPX Default Route and may not be used as an IPX network number. The set command will be ignored if a valid IPX network number is not configured.

Valid Values: X'0' to X'FFFFFFFD'

Default Value: 1

Example:
```
set net-number
IPX circuit number [1]? 2
IPX network number in hex
(0 is allowed only on IPXWAN unnumbered circuits) [1]?
```

node-id *network#*

Specifies the IPXWAN internal network number. A value of 0, FFFFFFFF or FFFFFFFE is not valid for the internal network number. IPXWAN will not be enabled unless a valid node ID is configured.

Default Value: 1

Example:
```
set node-id 2
```

remote-cache size *size*

Specifies the size of the remote cache routing table.

The size of the remote cache should equal the total number of remote networks used by the router plus a 10% buffer to prevent excessive purge requests.

Valid Values: The range is 1 to 10000.

Default Value: 64.

Example:
```
set remote-cache size
New IPX remote network cache size [64]? 80
```

rip-update-interval *ipx-circuit# interval*

Specifies the interval in minutes at which RIP periodic broadcasts should occur on a specific IPX circuit.

Increasing the RIP interval reduces traffic on WAN lines and dial circuits. It also prevents dial-on-demand circuit from dialing out so often.

Note: While complete RIP advertisements are controlled by the interval, the router still propagates network topology changes as quickly as it learns about them.

ipx-circuit#
Specifies an existing IPX broadcast to IPXWAN point-to-point circuit.

Valid Values: any valid IPX circuit number

Default: 1

interval
Specifies the interval in minutes

Valid Values: The range is from 1 to 1440 minutes.

Default Value: 1 minute. For additional information on RIP interval, see "Specifying RIP Update Interval" on page 372.

Example: `set rip-update-interval`

```
IPX circuit number [1]? 2
RIP Timer Value (minutes) [1]? 2
```

sap-update-interval *ipx-circuit# interval*
Specifies the time delay in minutes at which SAP periodic broadcasts should occur on a specific IPX circuit.

Increasing the SAP interval reduces traffic on WAN lines and dial circuits. It also prevents dial-on-demand circuit from dialing out so often.

Note: While complete SAP advertisements are controlled by the interval, the router still propagates service changes as quickly as it learns about them.

ipx-circuit#
Specifies an existing IPX broadcast or IPXWAN point-to-point circuit.

Valid Values: any valid IPX number

Default: 1

interval
Specifies the interval in minutes.

Valid Values: The range is from 1 to 1440 minutes.

Default Value: 1 minute.

Example: `set sap-update-interval`

```
IPX circuit number [1]? 2
SAP Timer Value (minutes) [1]? 2
```

split-horizon *heuristic enabled disabled*
Specifies the type of split-horizon used on the IPX circuit.

If there is only a single Frame Relay VC on the circuit, split-horizon is enabled; otherwise split-horizon is disabled.

Generally, split-horizon should be set to *enabled*.

heuristic
Enables split-horizon on the IPX circuit, except for Frame Relay IPX broadcast circuits.

Valid Values: any valid IPX circuit number

Default: 1

enabled
Enables split-horizon on the IPX circuit.

Valid Values: 1–1440

Default: 1

disabled
Disables split-horizon IPX circuit.

Valid Values: 1–1440

Default: 1

Example: `set split-horizon enabled 0`

```
IPX circuit number [1]? 2
```

Accessing the IPX Circuit Filter Configuration Environment

To access the IPX circuit Filter configuration environment, enter the following command at the `IPX config>` prompt:

```
IPX Config> filter-lists type
IPX type-List Config>
```

Where *type* is the type of IPX filter to be configured. Valid types are *router-lists, rip-lists, sap-lists,* and *ipx-lists*.

When creating a filter, an IPX circuit number is required.

IPX circuit Circuit-Filter Configuration Commands

This section describes the commands used to configure the IPX circuit-based filters; ROUTER, RIP, SAP, and IPX. To configure these filters, enter the `filter-lists type` command at the `IPX Config>` prompt, and then enter the configuration commands at the `IPX type-List Config>` prompt.

Table 61. IPX Filter Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Attach	Attaches a specified filter-list to a specified filter.
Create	Creates a filter or filter-list.
Default	Sets the default action of a filter to *include* or *exclude*
Delete	Deletes a filter or filter-list.
Detach	Detaches a filter-list from a filter.
Disable	Disables filtering.
Enable	Enables filtering.
List	Displays the current filtering configuration.

Table 61. IPX Filter Configuration Command Summary (continued)

Command	Function
Move	Reorders filter-lists attached to a filter.
Set-cache	Sets the caching size for a specified filter.
Update	Accesses the `IPX type-List filter-list Config>` prompt.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Attach

Use the **attach** command to attach a filter-list to a filter.

Syntax:

attach *list-name* *filter#*

list-name

Specifies the name of the filter-list. The **list** command can be used to display a list of the configured filter-list names.

Valid Values: Any alphanumeric string up to 16 characters

Default Value: None

filter#

Specifies the number of the filter. A numbered list of configured filters can be obtained using the list command.

Example: `attach test_list 1`

Create

Use the **create** command to create a filter-list or filter.

Syntax:

create list ...
filter ...

list *list-name*

Creates a list with the specified name.

Valid Values: Any alphanumeric string up to 16 characters

Default Value: none

You can also enter the **create list** command with no list name. You will then be prompted for the list name.

Example: `create list example_list`

filter *direction ipx-circuit#*

Creates a filter for the specified direction on the specified circuit. Specify *input* to filter packets received on the specified circuit. Specify *output* to filter packets to be sent by the specified circuit.

A number is automatically assigned to a filter when it is created and from that point on is used to identify the filter, rather than having to key in the circuit and direction (input or output) for all subsequent commands.

Example: `create filter input 1`

Default

Use the **default** command to set the default action for a filter. The default action is taken when no match is found for any of the filter items.

Syntax:

default *action filter#*

Example: `default exclude 1`

action
Specifies the default action. **Include** specifies that when no match is found to any of the filter items, the packet is processed. **Exclude** indicates that when no match is found, the packet is dropped.

filter#
Specifies the number of the filter. Use the **list** command to display a numbered list of configured filters.

Delete

Use the **delete** command to delete a filter-list or filter.

Syntax:

delete list ...
filter ...

list *list-name*
Deletes the specified list. The list command can be used to display the configured filter list names.

Example: `delete list example_list`

filter *filter#*
Deletes the specified filter. The list command can be used to display a numbered list of configured filters.

Example: `delete filter 1`

Detach

Use the **detach** command to detach a filter-list from a filter.

Syntax:

detach *list-name filter#*

list-name
Specifies the name of the filter-list. The list command can be used to display a list of the configured filter names.

Valid Values: Any alphanumeric string up to 16 characters

Default Value: None

filter#
Specifies the number of the filter. The list command can be used to display a numbered list of configured filters.

Example: `detach test_list 1`

Disable

Use the **disable** command to disable filtering globally or for a specified filter.

Syntax:

disable all
filter ...

all Disables all filters of the current type (ROUTER, RIP, SAP, or IPX).

Example: disable all

filter *filter#*
Disables the specified filter. Use the list command to display a numbered list of configured filters.

Example: disable filter 1

Enable

Use the **enable** command to enable filtering globally or for a specified filter.

Syntax:

enable all
filter ...

all Enables all filters of the current type (ROUTER, RIP, SAP, or IPX).

Example: enable all

filter *filter#*
Enables the specified filter. Use the list command to display a numbered list of configured filters

Example: enable filter 1

List

Use the **list** command to globally display the state of the current filtering type, or to display information about a specific filter.

Syntax:

list all
filter ...

all Lists information about the state of all filters of the current type.

Example: list all

```
Filtering: ENABLED

Filter Lists:
Name                           Action
------------------------------ ----------
ipx01                          EXCLUDE
ipx02                          INCLUDE
ipx03                          EXCLUDE

Filters:
Id    Circ  Ifc   Direction  State      Default    Cache
----- ----- ----- ---------- ---------- ---------- -----
1     3     2     INPUT      ENABLED    INCLUDE       10
2     2     1     INPUT      ENABLED    INCLUDE       10
```

filter *filter#*

Lists information about the specified filter. Use the list command to display a numbered list of configured filters.

Example: list filter 2

```
Filters:
Id    Circ  Ifc   Direction  State      Default    Cache
----- ----- ----- ---------- ---------- ---------- -----
2     2     1     INPUT      ENABLED    INCLUDE       10

Filter Lists:
Name                           Action
------------------------------ ----------
ipx01                          EXCLUDE
```

Move

Use the **move** command to change the order of filter lists within a filter. Packets are evaluated against the filter lists in the order the lists occur. The first match stops the filtering process.

Syntax:

move *src-list-name dst-list-name filter#*

src-list-name
Specifies the list to be moved within the filter.

dst-list-name
Specifies the list before which the src-list-name will be moved.

filter#
Specifies the filter to which the lists belong. The list command can be used to display a list of the configured filters and their attached filter lists.

Example: `move test-list-1 test-list-2 2`

Set-cache

Use the **set-cache** command to set the size of the filter cache. A filter cache is only supported for the IPX circuit filter; the ROUTER, RIP and SAP circuit filters do not support a cache.

Syntax:

set-cache *s*ize filter#

size
Specifies the size of the filter cache (in number of entries).

Valid Values: 4 to 64 cache entries.

Default Value: 10 entries.

filter#
Specifies the number of the filter. The list command can be used to display a numbered list of configured filters.

Example: `set-cache 10 1`

Update

The **update** command accesses the `IPX type-List list-name Config>` prompt. From this prompt you can issue commands to add, delete, or move items within the list being updated. From this prompt you can also set the action for the filter-list being updated.

Syntax:

update *list-name*

list-name
Specifies the name of the filter-list. The list command can be used to display the configured filter-list names.

Example: `update test-list`

Add (Update subcommand)

Use the **add** subcommand to add items to a filter-list. The list item parameters vary based on the type of circuit filter (ROUTER, RIP, SAP, or IPX) being configured. For all types of circuit filter, the **add** command can be entered without parameters. You will then be prompted for the required parameters.

Add (ROUTER)

Syntax:

add *node-number mask*

node-number
Specifies the value to be compared against the source node number of the router which sent the RIP response packet (after being ANDed with the mask). If you want to match on a single node, set the node-number parameter to the address and set the mask to FFFFFFFFFFFF. If you want to match on all nodes, set the node-number parameter and the mask parameter to 000000000000.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Default Value: none

mask
Specifies the value to be ANDed with the source node address of the router which sent the RIP response packet (before being compared with the address parameter).

If you want to match on a single address, set the address parameter to the address and set the mask to FFFFFFFFFFFF. If you want to match on all addresses, set the address parameter and the mask parameter to 000000000000.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Default Value: X'FFFFFFFFFFFF'

Example: `add 400000001000 ffffffff0000`

Add (RIP)

Syntax:

add *net-range-start net-range-end*

net-range-start
Specifies the start of a range (inclusive) of IPX network numbers to be filtered. If you want to match on a single network number, set the net-range-start and net-range-end parameters to that network number. If you want to match on all network numbers, set the net-range-start to X'00000001' and the net-range-end to X'FFFFFFFE'.

Valid Values: X'1' to X'FFFFFFFE'

Default Value: X'1'

net-range-end
Specifies the end of a range (inclusive) of IPX network numbers to be filtered.

Valid Values: X'1' to X'FFFFFFFE'

Default Value: X'1'

Example: `add 00000001 FFFFFFFE`

Add (SAP)

Syntax:

add *comparitor hops sap-type name*

comparitor
Specifies the type of hop count comparitor for this list item.

Valid Values:

- <
- <=
- =
- >=
- >

Default Value: <= The comparitor and hops parameters are ignored on output filters.

hops
Specifies the hop count for this list item. If you do not want to filter based on hop count, enter <= 16 for the comparitor and hop count. The comparitor and hops parameters are ignored on output filters.

Valid Values: 0 to 16

Default Value: 16

sap-type
Specifies the service type to be filtered. Enter the service type, or X'0000' for all service types.

Valid Values: X'0' to X'FFFF'

Default Value: 4

name
Specifies the service name to be filtered.

Valid Values:

A string of 1 to 47 ASCII characters (X'20' through X'7E').

The question mark (?) and asterisk (*) characters serve as wildcard characters. The question mark may be used multiple times to represent any single character within the server name. The asterisk may be used multiple times to represent any portion of the server name. The question mark and asterisk may also be used together.

Default Value: none

Example: `add < 6 0004 *`

Add (IPX)

Syntax:

add *comparitor hops ipx-type dst-net-range-start dst-net-range-end dst-node dst-mask dst-sck-range-start dst-sck-range-end src-net-range-start src-net-range-end src-node src-mask src-sck-range-start src-sck-range-end*

comparitor
Specifies the type of hop count comparitor for this list item. The comparitor and hops parameters are ignored on output filters.

Valid Values:
- <
- <=
- =
- >=
- >

Default Value: <=

hops
Specifies the hop count for this list item. If you do not want to filter based on hop count, enter <= 16 for the comparitor and hop count. The comparitor and hops parameters are ignored on output filters.

ipx-type
Specifies the IPX packet type to be filtered. Enter the packet type, or 00 for all packet types.

Valid Values: X'0' to X'FF'

Default Value: X'0'

dst-net-range-start
Specifies the start of a range (inclusive) of destination IPX network numbers to be filtered. If you want to match on a single network number, set the dst-net-range-start and dst-net-range-end parameters to that network number. If

you want to match on all network numbers, set the dst-net-range-start to X'00000001' and the dst-net-range-end to X'FFFFFFFE'.

Valid Values: X'00000000' to X'FFFFFFFF'

Default Value: X'00000000'

dst-net-range-end
Specifies the end of a range (inclusive) of destination IPX network numbers to be filtered. If you want to match on a single network number, set the dst-net-range-start and dst-net-range-end parameters to that network number. If you want to match on all network numbers, set the dst-net-range-start to X'00000001' and the dst-net-range-end to X'FFFFFFFE'.

Valid Values: X'00000000' to X'FFFFFFFF'

Default Value: X'00000000'

dst-node
Specifies the value to be compared against the destination node number (after being ANDed with the dst-mask). If you want to match on a single node, set the dst-node parameter to the node number and set the dst-mask to X'FFFFFFFFFFFF'. If you want to match on all nodes, set the dst-node parameter and the dst-mask parameter to X'000000000000'.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Default Value: X'000000000000'

dst-mask
Specifies the value to be ANDed with the destination node address (before being compared with the dst-address parameter). If you want to match on a single address, set the dst-address parameter to the address and set the dst-mask to X'FFFFFFFFFFFF'. If you want to match on all addresses, set the dst-address parameter and the dst-mask parameter to X'000000000000'.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Default Value: X'000000000000'

dst-sck-range-start
Specifies the start of a range (inclusive) of destination IPX sockets to be filtered. If you want to match on a single socket, set the dst-sck-range-start and dst-sck-range-end parameters to that socket. If you want to match on all sockets, set the dst-sck-range-start to X'0000' and the dst-sck-range-end to X'FFFF'.

Valid Values: X'0000' to X'FFFF'

Default Value: 0

dst-sck-range-end
Specifies the end of a range (inclusive) of destination IPX sockets to be filtered. If you want to match on a single socket, set the dst-sck-range-start and dst-sck-range-end parameters to that socket. If you want to match on all sockets, set the dst-sck-range-start to X'0000' and the dst-sck-range-end to X'FFFF'.

Valid Values: X'0000' to X'FFFF'

Default Value: 0

src-net-range-start
Specifies the start of a range (inclusive) of source IPX network numbers to be filtered. If you want to match on a single network number, set the src-net-range-start and src-net-range-end parameters to that network number. If you want to match on all network numbers, set the src-net-range-start to X'00000001' and the src-net-range-end to X'FFFFFFFE'.

Valid Values: X'00000000' to X'FFFFFFFE'

Default Value: X'00000000'

src-net-range-end
Specifies the end of a range (inclusive) of source IPX network numbers to be filtered. If you want to match on a single network number, set the src-net-range-start and src-net-range-end parameters to that network number. If you want to match on all network numbers, set the src-net-range-start to X'00000001' and the src-net-range-end to X'FFFFFFFE'.

Valid Values: X'00000000' to X'FFFFFFFE'

Default Value: X'00000000'

src-node
Specifies the value to be compared against the source node number (after being ANDed with the src-mask). If you want to match on a single node, set the src-node parameter to the node number and set the src-mask to X'FFFFFFFFFFFF'. If you want to match on all nodes, set the src-node parameter and the src-mask parameter to X'000000000000'.

Valid Values: X'00000000' to X'FFFFFFFF'

Default Value: X'00000000'

src-mask
Specifies the value to be ANDed with the source node address (before being compared with the src-address parameter). If you want to match on a single address, set the src-address parameter to the address and set the src-mask to X'FFFFFFFFFFFF'. If you want to match on all addresses, set the src-address parameter and the src-mask parameter to X'000000000000'.

Valid Values: X'000000000000' to X'FFFFFFFFFFFF'

Default Value: X'000000000000'

src-sck-range-start
Specifies the start of a range (inclusive) of source IPX sockets to be filtered. If you want to match on a single socket, set the src-sck-range-start and src-sck-range-end parameters to that socket. If you want to match on all sockets, set the src-sck-range-start to X'0000' and the src-sck-range-end to X'FFFF'.

Valid Values: X'0000' to X'FFFF'

Default Value: X'0000'

src-sck-range-end
Specifies the end of a range (inclusive) of source IPX sockets to be filtered. If you want to match on a single socket, set the src-sck-range-start and src-sck-range-end parameters to that socket. If you want to match on all sockets, set the src-sck-range-start to 0000 and the src-sck-range-end to FFFFF.

Valid Values: X'0000' to X'FFFF'

Default Value: X'0000'

Example:

```
add  <= 16  0  00000004 00000004 000000000000 000000000000
0000 FFFF 0000005A 0000006A 000000000000 000000000000 0000 FFFF
```

This example filters all packets from IPX networks 5A through 6A to IPX network 4.

Delete (Update subcommand)

Use the **delete** subcommand to delete an item from the current filter-list.

Syntax:

delete *item#*

item#
Specifies the number of the item in the list. The number can be obtained by using the list command to list the items in the filter-list.

Example: `delete 4`

List (Update subcommand)

Use the **list** subcommand to display the filter-list action and list filter items.

Syntax:

list

Example: `list`

```
IPX IPX-List 'ipx01' Config>list
Action: EXCLUDE
Id   Hops Type Net Range           Address      Mask         Sock Range
---- ---- ---- ------------------- ------------ ------------ -----------
1    <=16    0     4320 -     4324 4000003A0002 FFFFFFFFFFFF    0 - FFFF (Dest)
                   3A33 -    13A33 400000010000 FFFFFFFF0000    0 - FFFF (Source)
```

Move (Update subcommand)

Use the **move** subcommand change the order of filter items. After you change the order of filter items, they are renumbered to reflect the new order. The list command can be used to display a numbered list of configured filter items.

The *src-line#* parameter indicates the line to be moved. This line will be moved to precede the item specified by the *dest-line#* parameter.

Syntax:

move *src-line# dest-line#*

Example: `move 5 2`

Set-action (Update subcommand)

Use the **set-action** subcommand to indicate the action to be taken when a match is made to a filter-list

Syntax:

set-action include
exclude

include

Specifies that if a match is found for the current filter, the packet will be processed (included) for ROUTER and IPX filters. For RIP and SAP filters, **include** specifies that the RIP or SAP entry will be processed.

Example: `set-action include`

exclude

Specifies that if a match is found for the current filter, the packet will be dropped (excluded) for ROUTER and IPX filters. For RIP and SAP filters, **exclude** specifies that if a match is found, the RIP or SAP entry will be ignored.

Example: `set-action exclude`

Accessing the IPX Monitoring Environment

For information on how to access the IPX monitoring environment, refer to “Getting Started (Introduction to the User circuit)” in the *8371 Interface Configuration and Software User’s Guide*

IPX Monitoring Commands

Table 62 lists the IPX monitoring commands. The IPX monitoring commands allow you to view the parameters and statistics of the circuits and networks that transmit IPX packets. Monitoring 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 IPX monitoring commands at the `IPX>` prompt. Table 62 summarizes the IPX monitoring commands.

Table 62. IPX Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Access-controls	Displays whether the global IPX filter (access control) is enabled, the IPX access-control statements, and the number of packets that have matched each access-control statement.
Cache	Lists the current contents of the routing cache.
Counters	Displays the number of routing errors and packet overflows.
Delete keepalive connection	Deletes a Keepalive filtering table entry.

Table 62. IPX Monitoring Command Summary (continued)

Command	Function
Disable	Disables IPX globally or on specific IPX circuits.
Dump routing tables	Displays the contents of the routing table.
Enable	Enables IPX globally or on specific IPX circuits.
Filters	Displays whether global SAP filtering is enabled, the SAP filter statements, and a count of the SAP advertisements which have been filtered.
Filter-Lists	Accesses the IPX circuit filter console. This is where the RIP router, RIP SAP, and IPX circuit-based filters can be monitored.
Keepalive	Displays the status of each active client/server connection in the keepalive-filtering table.
List	Lists the current configuration or the IPX address of each enabled circuit.
Ping	Sends IPXPING packets to another host and watches for a response. This command can be used to isolate trouble in an internetwork environment.
Recordroute	Sends IPXPING record route packets to another host and watches for a response. Use this command to record and display the round-trip route between this device and another host. Use this information to isolate trouble in an internetwork environment.
Reset	Resets specific IPX circuits, global SAP filters, global IPX filters (access controls), static routes, static services, or the router, RIP, SAP, or IPX circuit-based filters (filter lists).
Sizes	Displays the configured sizes of the local node and remote network caches, and the number of cache entries currently in use.
Slist	Displays the contents of the IPX SAP server table.
Traceroute	Sends IPXPING trace route packets to another host and watches for a response. Use this command to trace and display each hop a packet takes on its way from this device to a destination host. Use this information to isolate trouble in an internetwork environment.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Access Controls

Use the **access-controls** command to list the status of global IPX filters (access controls), the IPX access control statements, and a count of how many times each control statement has been followed.

Syntax:

access-controls

Example: `access-controls`

```
IPX Access Controls:  Enabled
#   T Dest Net Host         Sock Sock Src Net  Host         Sock  Sock Count
1   E 2        000000000000 0    FFFF 3        000000000000 0     FFFF 0
2   I 0        000000000000 452  453  0        000000000000 0     FFFF 0
```

Access control index number

Type Identifies whether packets are sent or dropped for a specific address or set of addresses. I means include. This allows the packets to be sent. E means exclude. This causes the router to discard the packets.

Dest-net
Network number of the destination. Zero (0) means all networks.

Dest-host
Host number on the destination network (0) means all hosts on the network.

Dest-sck
Two numbers that specify an inclusive range of destination sockets.

Src-net
Network number of the source. Zero (0) means all networks.

Src-host
Host number on the source network. Zero means all hosts on the network.

Src-sck
Two numbers that specify an inclusive range of source sockets.

Count Specifies the number of incoming IPX packets that have matched each access-control statement, causing the associated Type (Include or Exclude) to be performed.

Cache

Use the **cache** command to display the contents of the IPX routing cache.

Syntax:

cache

Example: `cache`

```
Dest Net/Node          Use Count      via Net/Node          Circ  Ifc
     420               1                  412/000004200000  3     2
     412               1                  412/000000000000  3     2
     412/000004200000  1                  412/000004200000  3     2
```

The first entry shows that the remote network 420 can be reached over the serial circuit with IPX network number 412. The second entry is the IPX network 412. It is an Ethernet directly attached to the router. This entry is a general local network entry. There will be one general local network entry for each of the directly attached networks after they have begun forwarding IPX packets. The last entry is a local entry on an Ethernet. This IPX cache entry has been used to send 1 packet to the IPX node number 0000 0420 0000 on net number 412.

Counters

Use the **counters** command to display the number of routing errors and packet overflows that have occurred. In the example, the counters show no recorded errors.

Syntax:

counters

Example: `counters`

```
Routing errors
Count  Type
0      Unknown
0      Checksum error
0      Destination unreachable
0      Hop count expired
0      circuit size exceeded

Destination errors
Count  Type
```

```
0      Unknown
0      Checksum error
0      Non-existent socket
0      Congestion

IPX input packet overflows
Circ   Ifc    Name        Count
1      0       Eth/0      0
2      1       PPP/0      0
3      2       PPP/1      0
```

Routing Errors

Unknown
An unspecified error occurred before reaching the destination.

Checksum
The checksum is incorrect, or the packet had some other serious inconsistency before reaching the destination.

Destination unreachable
The destination host cannot be reached from here.

Hop count expired
The packet has passed through 15 internet routers without reaching its destination.

circuit size exceeded
The packet is too large to be forwarded through some intermediate network.

Destination errors

Unknown
An unspecified error was detected at destination.

Checksum
The checksum is incorrect, or the packet has some other serious inconsistency detected at destination.

Nonexistent socket
The specified socket does not exist at the specified destination host.

Congestion
The destination cannot accept the packet due to resource limitations.

IPX Input Packet Overflows

Net Specifies the circuit name.

Count Specifies the number of packets that could not be received due to resource limitations.

Delete

Use the **delete** command to remove a Keepalive filtering table entry.

Syntax:
delete *entry#*

entry# Specifies the table entry to be deleted. The **Keepalive** command can be used to list the contents of the Keepalive filtering table.

Example: `delete 1`

Disable

Use the **disable** command to disable IPX globally or on specific circuits.

Syntax:

disable circuit ...
ipx

circuit *ipx-circuit#*
Disables the IPX circuit specified by *ipx-circuit#*. IPX can be re-enabled using the **enable** command.

Example: `disable circuit 2`

ipx Disables IPX globally on all IPX circuits. IPX can be globally re-enabled using the **enable** command.

Example: `disable ipx`

Dump

Use the **dump** command to display the contents of the routing tables.

Syntax:

dump

Example: `dump`

```
Type Dest Net Hops Delay Age(M: S)   via Router        Circ  Ifc
Dir  412      0    6        0: 0      412/000004000000 3     2
Dir  400      0    1        0: 0      400/020000000400 1     0
Dir  411      0    3        0: 0      411/400000000400 2     1
Stat 1        3    2        0: 0      400/010101010101 1     0
RIP  420      1    7        0:30      412/000004200000 3     2
Stat 444      2    2        0: 0      400/400000000444 1     0
Stat FFFFFFFD 14   3000     0: 0      400/111111111111 1     0
```

Type

- Dir - specifies that this network is directly connected to the router.
- RIP - specifies that this route was provided by the IPX routing protocol, RIP.
- Old - specifies that this route has timed out and is no longer being used. The route remains in the table briefly to inform other routers that the route is no longer valid; after this brief interval, it is no longer displayed.
- Stat - specifies that this is a static route.

Dest net
Specifies the destination network number.

Hops Specifies the number of hops to this destination.

Delay Specifies the estimate of how long it takes the router to transmit and for the packet to arrive at its destination. The unit of delay is the number of IBM PC clock ticks to send a 576-byte packet, which is 18.21 clock ticks per second. The minimum delay is 1 unit.

Age Specifies the age of the routing information in minutes and seconds. If an entry in the routing table is not updated, the router takes the following actions:

- After three RIP update intervals have passed, the route is specified as Old and the router advertises that the route is no longer valid. The RIP

update interval can be displayed using the IPX **config** command. For additional information on RIP intervals, see “Specifying RIP Update Interval” on page 372.

- After an additional 60 seconds, the route is deleted and does not appear in the dump display.

Via router
Specifies the next hop for packets going to networks that are not directly connected. For directly connected networks, this is the address of the router circuit that transmits the packet.

Circ IPX circuit number

Ifc Network interface number

At the top of the display is the number of route and network entries used and the total available. If all the network entries are used, it is likely that the routing table is not large enough. Use the IPX configuration **set maximum networks** command to increase the size.

If all of the route entries are used, then there may be routes to IPX networks that cannot be kept, including new, incoming networks. If you do not want to increase the number of available routes, reduce the number of maximum routes per network.

Enable

Use the **enable** command to enable IPX globally or on specific circuits.

Syntax:
enable circuit ...
ipx

circuit *ipx-circuit#*
Enables IPX on the circuit specified by *ipx-circuit#*. An IPX network number must have been configured for the circuit before IPX can be enabled.

Example: `enable circuit 2`

ipx Globally enables IPX on all enabled IPX circuits.

Example: `enable ipx`

Filters

Use the **filters** command to display whether global SAP filtering is enabled, the SAP filter statements, and a count of the SAP advertisements that have been filtered.

Syntax:
filters

Example: `filters`

```
IPX SAP Filters:  Enabled
Count  Max Hops  Type  Service Name
0      5         4     FILESRV01
```

Count Indicates the number of SAP advertisements that have been filtered (discarded).

Max Hops
Indicates the maximum number of hops permitted for the service.

Type Is the numeric service class.

Service name
Is the name of the service if it has a name.

Filter-lists

Use the **filter-lists** command to access the `IPX type-Lists>` prompt. Valid types are: router-lists, rip-lists, sap-lists, and ipx-lists.

For information about the commands available from this prompt, see “IPX Circuit Filter Monitoring Commands” on page 432.

Syntax:

filter-lists	router-lists
	rip-lists
	sap-lists
	ipx-lists

Example: `filter-lists router-lists`

Keepalive

Shows the status of each active client/server connection in the keepalive-filtering table.

Syntax:

keepalive

Example:

```
Keepalive
Conn #     Net  /   Node     /Sock        Net  /   Node     /Sock
          ------------------------------------------------------------------
0         272727/000000000001/4001         302/0000C911EF1C/4004
               (server conn # 1, conn type: passive, last heard  1:00 ago)
1         272727/000000000001/4001         302/0000C911B0D9/4004
               (server conn # 2, conn type: passive, last heard  1:00 ago)
```

List

Use the **list** command to list the current configuration or the IPX address of each enabled IPX circuit.

Syntax:

list	*addresses*
	configuration

addresses
Lists the IPX address of each enabled IPX circuit.

Example:

```
Circ   Ifc    Name        Type                          Network/Address
1      0       Eth/0      Ethernet                      400/020000000400
```

Configuration

List the current IPX configuration. This command displays the same information as the **list summary** configuration command. See “List” on page 398 for an example of the display and an explanation of the output.

Ping

Use the **ping** command to make the router send IPXPING packets to a given destination (“pinging”) and watch for a response. This command can be used to isolate trouble in an internetwork environment.

This process is done continuously. Matching received responses are displayed with the sender’s IPX network number and node number, the number of hops, and the round-trip time in milliseconds.

To stop the pinging process, type any character at the monitoring. At that time, a summary of packet loss, round-trip time, and number of unreachable destinations will be displayed.

When a multicast address is given as destination, there may be multiple responses for each packet sent, one for each group member. Each returned response is displayed with the source address of the responder.

Notes:

1. Care should be taken when specifying the broadcast address (FFFFFFFFFFFF), as this could generate a large number of IPXPING response packets, which would degrade network and routing software performance.
2. If you enter the **ping** command without any parameters, you will be prompted for all parameters. If you enter only ***destination network*** and ***destination node***, default values will be used for the remaining parameters.

Syntax:

ping *dest-net dest-node src-net src-node size rate*

dest-net

Specifies the destination IPX network number. This parameter is required.

Valid Values: X'1' to X'FFFFFFFD'

Default Value: 1

dest-node

Specifies the destination IPX node address. This parameter is required.

Valid Value: X'1' to X'FFFFFFFFFFFF'

Default Value: None

src-net

Specifies the source IPX network number. This is an optional parameter. The value must be a known network number that is associated with a direct attached IPX circuit. If a source network is not specified, the network number of the IPX circuit on which the IPXPING request packets are sent will be used as

the source IPX node. If the IPX circuit is an IPXWAN unnumbered RIP or static routing circuit, the node address of the IPX circuit used for the source network number will be used as the source node.

Valid Value: X'1' - X'FFFFFFFD'

Default Value: 1

src-node
Specifies the source IPX node address. This is an optional parameter. The value must be a known node address that is associated with a direct attached IPX circuit. If a source node is not specified, the node address of the IPX circuit on which the IPXPING request packets are sent will be used as the source IPX node. If the IPX circuit is an IPXWAN unnumbered RIP or static routing circuit,, the node address of the IPX circuit used for the source network number will be used as the source node.

Valid Value: X'1' - X'FFFFFFFFFFFE'

Default Value: None

size
Specifies the number of data bytes to be appended to the ping request. This is an optional parameter. The data includes the time the request is first sent so the amount specified cannot be smaller than 4 bytes. It also cannot be larger than the maximum packet size supported by the router or the output circuit. This value can vary depending on the configuration.

Valid Value: 4 to Router Maximum

Default Value: 56 bytes

rate
Specifies the number of seconds between ping requests. This is an optional parameter.

Valid Value: 1 to 60

Default Value: 1

Example: ping

```
Destination network number [1]? 20
Destination node number []? 00000001c200
Source network number [1]? 10
Source node number []? 000000019a00
Data size: [56]?
Rate in seconds [1]?

IPXPING 20/00000001C200: 56 data bytes
56 data bytes from 20/00000001C200: hops=3 time=0 ms
56 data bytes from 20/00000001C200: hops=3 time=40 ms
56 data bytes from 20/00000001C200: hops=3 time=0 ms

----20/00000001C200 IPXPING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms) min/ave/max = 0/13/40
```

RecordRoute

Use the **recordroute** command to report every forwarding circuit on the path to the destination and back again. If recordroute is invoked with no parameters, you will

be prompted for all of them. Only the destination IPX network number and destination IPX node address are required.

There are two events that will end a recordroute. The first is when you press a key. The second is when the maximum number of recordroute request packets have been sent.

Syntax:

recordroute *dest-net dest-node src-net src-node rate number*

dest-net
Specifies the destination IPX network number. This parameter is required.

Valid Values: X'1' to X'FFFFFFFD'

Default Value: 1

dest-node
Specifies the destination IPX node address. This parameter is required.

Valid Values: X'1' to X'FFFFFFFFFFFE'

Default Value: None

src-net
Specifies the source IPX network number. This is an optional parameter. The value must be a known network number that is associated with a direct attached IPX circuit. If a source network is not specified, the network number of the IPX circuit on which the recordroute packets are sent will be used as the source IPX address. If the IPX circuit is an IPXWAN unnumbered RIP or static routing circuit, the network number of some other numbered IPX circuit will be used as the source address, since IPXWAN unnumbered RIP and static routing circuits are not assigned an IPX network number.

Valid Values: X'1' to X'FFFFFFFD'

Default Value: 1

src-node
Specifies the source IPX node address. This is an optional parameter. The value must be a known node address that is associated with a direct attached IPX circuit. If a source node is not specified, the node address of the IPX circuit on which the recordroute packets are sent will be used as the source IPX node. If the IPX circuit is an IPXWAN unnumbered RIP or static routing circuit, the node address of IPX circuit used for the source network number will be used as the source node.

Valid Values: X'1' to X'FFFFFFFFFFFE'

Default Value: None

rate
Specifies the number of seconds between recordroute requests. This is an optional parameter.

Valid Values: 1 to 60

Default Value: 1

number
Specifies the maximum number of recordroute requests to be sent. This is an optional parameter. A value of zero will cause the recordroute to continue until a key is pressed.

Valid Values: 0 to 60

Default Value: 0

Example: recordroute

```
Destination network number [1]? 20
Destination node number []? 00000001c200
Source network number [1]? 10
Source node number []? 000000019a00
Rate in seconds [1]?
Number of packets to send [0]?

RECORDROUTE 20/00000001C200: 784 data bytes
784 data bytes from 20/00000001C200: seq_no=0 time=0 ms
Recorded Routes (in hex):
                10/000000019A00
                500/0000100A0000
                500/0000100C0000
                10/000000019000
                10/000000019A00 (Final Destination)

784 data bytes from 20/00000001C200: seq_no=1 time=30 ms  (same route)
784 data bytes from 20/00000001C200: seq_no=2 time=10 ms  (same route)
...
784 data bytes from 20/00000001C200: seq_no=18 time=0 ms
Recorded Routes (in hex):
                10/000000019A00
                0/0000100A0000
                20/00000001AE00
                20/00000001C200
                0/0000100B0000
                10/000000019000
                10/000000019A00 (Final Destination)

784 data bytes from 20/00000001C200: seq_no=19 time=0 ms  (same route)
784 data bytes from 20/00000001C200: seq_no=20 time=70 ms  (same route)
784 data bytes from 20/00000001C200: seq_no=21 time=0 ms  (same route)
...
784 data bytes from 20/00000001C200: seq_no=48 time=0 ms
Recorded Routes (in hex):
                10/000000019A00
                500/0000100A0000
                500/0000100C0000
                10/000000019000
                10/000000019A00 (Final Destination)

784 data bytes from 20/00000001C200: seq_no=49 time=0 ms  (same route)
784 data bytes from 20/00000001C200: seq_no=50 time=0 ms  (same route)

----20/00000001C200 RECORDROUTE Statistics----
53 packets transmitted, 38 packets received, 28% packet loss
5 unreachables, 0 no usable source addresses, 0 buffer unavailable
round-trip (ms) min/ave/max = 0/23/100
```

The entire path is reported only once on the first response or when the path changed. In the above example, the path changed twice.

Reset

Use the **reset** command to reset specific IPX circuits, global SAP filters, global IPX filters (access controls), static routes, static services, or the Router, RIP, SAP, or IPX circuit-based filters (filter lists).

Syntax:

reset access-controls
circuit . . .
filters
filter-lists
route-static
sap-static

access-controls
Resets the global IPX filters (access-controls) based on the configuration parameter stored in the configuration memory. Changes made to the global IPX filter configuration will be activated.

Example: reset access-controls

circuit *ipx-circuit#*
Resets IPX on the specified IPX circuit using configuration parameter values stored in the configuration memory. Changes made to the IPX configuration on the IPX circuit will be activated.

Example: reset circuit 2

filters
Resets the global SAP filters based on the configuration parameter values stored in the configuration memory. Changes made to the global SAP filter configuration will be activated.

Example: reset filters

filter-lists *filter-type*
Resets the circuit-based filter based on configuration parameter values stored in the configuration memory. Changes made to the circuit-based filter configuration will be activated. Valid ***filter-types*** are router,rip,sap, and ipx.

Example: reset filter-lists rip

route-static
Resets the static routes based on the configuration parameter values stored in the configuration memory. Changes made to the static route configuration will be activated.

Example: reset route-static

sap-static
Resets the static services based on the configuration parameter values stored in the configuration memory. Changes made to the static services configuration will be activated.

Example: reset sap static

Sizes

Use the **sizes** command to display the configured sizes of the local node and remote network caches, and the number of cache entries currently in use. (This command does not display the contents of the caches.)

Syntax:

sizes

Example: `sizes`

```
Current IPX cache size:
Remote network cache size (max entries): 64
       2 entries now in use

Local node cache size (max entries): 128
       1 entries now in use
```

Slist

Use the **slist** command to display the contents of the IPX SAP server table.

Syntax:

slist

Example: `slist`

```
9 entries used out of 32

State Typ Service Name                  Hops   Age     Net  /   Host     /Sock
SAP  4    PCS12                          3     0:50        1/000000000048/0451
SAP  4    ACMPCS                         3     0:50        1/00000000004A/0451
SAP  4    DEVEL2                         1     0:50       11/0000000000B4/0451
SAP  4    PLANNING                       2     0:50       BB/0000000000B7/0451
SAP  4    DEVEL                          2     0:50       BB/0000000000EE/0451
SAP  4    SOFT2                          1     0:30      704/000000000094/0451
SAP  4    SKYSURF1                       2     0: 5 2C39ABE9/000000000001/0451
SAP  278  DIRTREE                        2     0: 5 2C29ABE9/000000000001/4005
Stat 26B  DIRTREE                        2     0: 0      444/000000000001/0045
```

State Specifies one of the following parameters:

SAP - indicates that this service was obtained by the SAP routing protocol.

Del - indicates that this service has timed out and is no longer being used. The service is kept briefly in the table to inform other routers that the service is no longer valid. After that, it is deleted and is no longer displayed.

Stat - indicates that this service is a static service.

Typ Specifies the server type in hexadecimal. File servers are type 0004. Other type numbers are assigned by Novell.

Service name

Specifies the server's unique name for this type of server. Only the first 30 characters of the 47-character name are displayed to conserve space.

Hops Specifies the number of router hops from this router to the server.

Age Specifies the age of the service information. If an entry in the SAP table is not updated, the router takes the following actions:

- After 3 SAP update intervals have passed, the service is specified as Del and the router advertises that the service is no longer valid. The SAP update interval can be displayed using the IPX **config** command.

- After an additional 60 seconds, the service is deleted and does not appear in the **slist** display.

Net/Host/Sock
Specifies the address of the service. The address includes the following parameters:

- Network number
- Net host number (the address of the first circuit on the network)
- Socket number at which the service can be reached

At the bottom of the display is the number of entries used and the total available. If all the entries are used, it is likely that the service table is not large enough. Use the IPX configuration **set maximum services** command to increase the size.

Traceroute

Use the **traceroute** command to report each hop a ping request takes on its way to a final destination. If traceroute is invoked with no parameters, you will be prompted for all of them. Only the destination IPX network number and destination IPX node address are required.

There are three events that will end a traceroute. The first is when you press a key. The second is when a response is received from the destination address. The third is when the maximum number of hops has been reached.

Syntax:

traceroute *dest-net dest-node src-net src-node size probes rate hops*

dest-net
Specifies the destination IPX network number. This parameter is required.

Valid Values: X'1' to X'FFFFFFFD'

Default Value: 1

dest-node
Specifies the destination IPX node address. This parameter is required.

Valid Values: X'1' to X'FFFFFFFFFFFE'

Default Value: None

src-net
Specifies the source IPX network number. This is an optional parameter. The value must be a known network number that is associated with a direct attached IPX circuit. If a source network is not specified, the network number of the IPX circuit on which the traceroute packets are sent will be used as the source IPX address. If the IPX circuit is an IPXWAN unnumbered RIP or static routing circuit, the network number of some other numbered IPX circuit will be used as the source address, since IPXWAN unnumbered RIP and static routing circuits are not assigned an IPX network number.

Valid Value: X'1' to X'FFFFFFFD'

Default Value: 1

src-node
Specifies the source IPX node address. This is an optional parameter. The value must be a known node address that is associated with a direct attached IPX circuit. If a source node is not specified, the node address of the IPX circuit on which the traceroute packets are sent will be used as the source IPX node. If the IPX circuit is an IPXWAN unnumbered RIP or static routing circuit, the node address of IPX circuit used for the source network number will be used as the source node.

Valid Values:X'1' to X'FFFFFFFFFFFE'

Default Value: None

size
Specifies the number of data bytes to be appended to the traceroute request. This is an optional parameter. The data includes the time the request is first sent, so the number specified cannot be smaller than 4 bytes. It also cannot be larger than the maximum packet size of the router or the output circuit. This value can vary depending on the configuration.

Valid Values: 4 to router maximum

Default Value: 56

probes
Specifies how many traceroute requests to send per hop. This is an optional parameter.

Valid Values: 1 to 10

Default Value: 3

rate
Specifies the number of seconds to wait between probes, when there is not an answer to the traceroute request. This is an optional parameter.

Valid Values: 1 to 60

Default Value: 1

hops
Specifies the maximum number of hops to send traceroute requests. This is an optional parameter. Without NLSP, a packet can traverse a maximum of 16 nodes (hence the default of 16). With NLSP or the IBM 6611 half-router solution, the limit is no longer 16.

Valid Values: 1 to 255

Default Value: 16

Example: `traceroute`

```
Destination network number [1]? 20
Destination node number []? 00000001c200
Source network number [1]? 10
Source node number []? 000000019a00
Data size:  [56]?
Number of probes per hop [3]?
Wait time between retries in seconds [1]?
Maximum Hops [16]?
```

```
TRACEROUTE 20/00000001C200: 56 data bytes
1 10/000000019000: 0 ms * 500/0000100B0000 20 ms
2 * * *
3 20/00000001C200: 10 ms 60 ms 20 ms
```

The source IPX address of a traceroute response is reported only once as long as it does not change. In the above example, two different routers responded to the one hop traceroute request. This would happen if the route to the destination changed between probes.

There is other information reported by traceroute besides the round trip time of a probe:

- '*' - No response packet was received in the time specified.
- 'H!' - The destination network is unreachable. This would be reported if the route to the destination was lost after traceroute was started.
- 'BF' - No buffers available.

IPX Circuit Filter Monitoring Commands

Table 63 lists the commands available from the IPX *type*-Lists> prompt. Each of these commands is explained in detail in this section.

To access the IPX *type*-Lists> prompt, enter **filter-lists** *type* at the IPX> prompt. Valid types are router-lists, rip-lists, sap-lists, and ipx-lists.

Table 63. IPX circuit Filter Command Summary

Command	Function
Cache	Displays the contents of the filter cache for the specified circuit. Only the IPX filter supports a filter cache.
Clear	Clears the counters of the specified filter, or clears the counters of all filters of the current type (ROUTER, RIP, SAP, or IPX).
Disable	Disables a specified filter, or all filters of the current type.
Enable	Enables a specified filter, or all filters of the current type.
List	Lists a specified filter, or all filters of the current type.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Cache

Use the **cache** command to display the contents of the filter cache. Only the IPX filter supports a cache. ROUTER, RIP, and SAP filters do not support a filter cache.

Syntax:

cache filter *filter#*

filter# Specifies the number of the filter. The list command can be used to display a numbered list of configured filters.

Example: `cache filter 1`

```
IPX IPX-Lists>cache filter 1
Hops Type Dst Net  Address      Sock Src Net  Address      Sock Action
---- ---- -------- ------------ ---- -------- ------------ ---- -------
   4   00 04000000 400003900000  802 03000040 400003004400  966 EXCLUDE
   2   00 0004A300 400000233D00  952 0763A020 4000000DD100  920 INCLUDE
```

Clear

Use the **clear** command to clear the counters of the specified filter, or to clear the counters of all filters of the current type (ROUTER, RIP, SAP, or IPX).

Syntax:

clear all
filter ...

all Clears the counters of all filters of the current type (ROUTER, RIP, SAP, or IPX).

Example: `clear all`

filter *filter#*
Clears the counters of the specified filter number. The list command can be used to display a numbered list of configured filters.

Example: `clear filter 1`

Disable

Use the **disable** command to disable specific filters or to disable all filters of the current type (ROUTER, RIP, SAP, or IPX).

Syntax:

disable all
filter *filter#*

all Disables all filters of the current type (ROUTER, RIP, SAP, or IPX).

Example: `disable all`

filter *filter#*
Disables the specified filter number. The list command can be used to display a numbered list of configured filters.

Example: `disable filter 1`

Enable

Use the **enable** command to enable specific filters or to enable all filters of the current type (ROUTER, RIP, SAP, or IPX).

Syntax:

enable all
filter *filter#*

all Enables all filters of the current type (ROUTER, RIP, SAP, or IPX).

Example: `enable all`

filter *filter#*
Enables the specified filter number. The list command can be used to display a numbered list of configured filters.

Example: `enable filter 1`

List

Use the **list** command to display information about specific filters, or about all filters of the current type (ROUTER, RIP, SAP, or IPX).

Syntax:

list all

filter *filter#*

all Lists the configuration of all filters of the current type (ROUTER, RIP, SAP, or IPX).

Example: `list all`

```
IPX IPX-Lists>list all
Filtering: ENABLED

Filter Lists:
Name                           Action
------------------------------ ----------
ipx01                          EXCLUDE
ipx02                          INCLUDE
ipx03                          EXCLUDE

Filters:
Id    Circ  Ifc   Direction  State      Default    Cache
----- ----- ----- ---------- ---------- ---------- -----
1     1     0     INPUT      ENABLED    INCLUDE    10
2     1     0     OUTPUT     ENABLED    INCLUDE    10
3     2     1     INPUT      DISABLED   INCLUDE    10
4     2     1     OUTPUT     DISABLED   INCLUDE    10
```

filter *filter#*

Lists the configuration of the specified filter number. The list command can be used to display a numbered list of configured filters.

Example: `list filter 1`

```
IPX IPX-Lists>list filter 1

Filters:
Id    Circ  Ifc   Direction  State      Default    Cache
----- ----- ----- ---------- ---------- ---------- -----
1     1     0     INPUT      ENABLED    INCLUDE    10

Filter Lists:
Name                           Action     Count
------------------------------ ---------- ----------
ipx01                          EXCLUDE    43
ipx02                          INCLUDE    23453
```

Chapter 33. Using ARP

This chapter describes how to use the Address Resolution Protocol (ARP) and the Inverse Address Resolution Protocol (Inverse ARP) on your router. It includes the following sections:

- "ARP Overview"
- "Inverse ARP Overview" on page 436

Note: If the device's software load does not contain Asynchronous Transfer Mode (ATM), ATM-related commands are not valid and are not displayed at the ARP configuration and console prompts.

ARP Overview

The ARP Protocol is a low-level protocol that dynamically maps network layer addresses to ATM addresses or physical medium access control (MAC) addresses. Given only the network layer address of the destination system, ARP locates the ATM address or MAC address of the destination host within the same network segment.

For example, a router receives an IP packet destined for a host connected to one of its LANs. The packet contains only a 32-bit IP destination address. To construct the data link layer header, a router acquires the physical MAC address of the destination host. Then, the router maps that address to the 32-bit IP address. This function is called *address resolution*. Figure 29 on page 436 illustrates how ARP works.

© Copyright IBM Corp. 1996, 1999

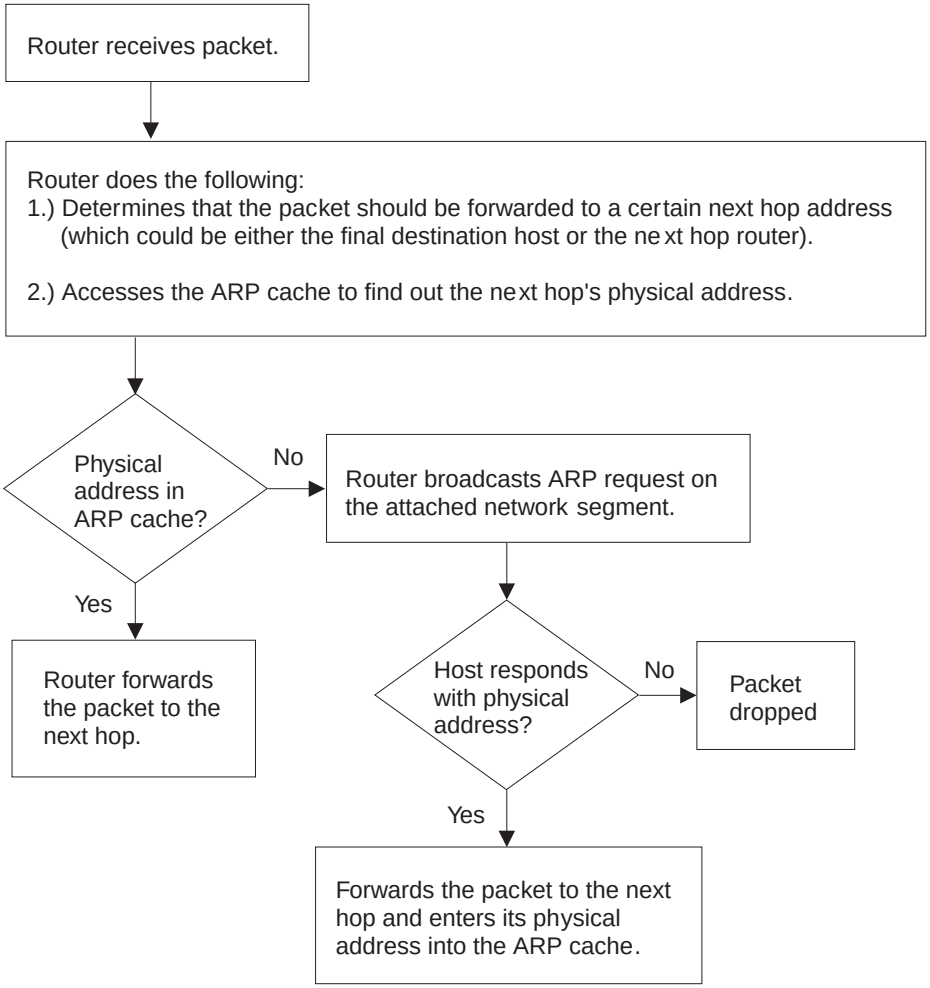

Figure 29. ARP Address Resolution Broadcast

When a router translates a network layer address to a physical address, the router accesses the ARP (translation) cache. The ARP cache contains the physical MAC address that corresponds to that network layer address. If the address is missing, the router broadcasts an ARP request to all hosts on the attached network segment to locate the correct physical MAC address. The node with the correct physical MAC address responds to the router. The router then sends the packet to the node and enters the physical MAC address into the translation cache for future use.

Inverse ARP Overview

Inverse ARP, described in RFC 1293/2390, was created for Frame Relay networks. This protocol defines a method for routers on a Frame Relay network to learn the protocol addresses of other routers in a way that very efficiently reduces traffic by eliminating the need to use broadcast ARP packets for address resolution. Inverse ARP discovers a protocol address by sending Inverse ARP request packets to the hardware address (for Frame Relay circuits the circuit identifier is the Frame Relay equivalent of a hardware address; for ATM, an ATM address is exchanged), as soon as the circuit becomes active. The remote router responds with its protocol address and the resulting mapping is stored in the ARP cache.

In ATM, the inverse ARP packet has been extended to handle the variable-sized ATM addresses of the source and destination. Addresses learned by inverse ARP are aged out in the same way as those learned by ARP.

The protocol address-to-hardware address entries learned by Inverse ARP do not time out when the ARP refresh timer expires. The mappings do not age at all except when the Frame Relay circuit goes down. This means that the router does not need to transmit any ARP broadcasts to update the ARP cache. However, the router permits updates to an entry when the other (remote) router changes its protocol address.

Support for both ARP and Inverse ARP greatly enhances the router's interoperability with other vendors' routers over Frame Relay for dynamic mapping of protocol and hardware addresses. If other Frame Relay-attached routers support Inverse ARP, then the mappings are dynamically learned as described above. If the attached routers do not support Inverse ARP but support "traditional" ARP on Frame Relay, then the mappings still could be learned dynamically using ARP exchanges (see Figure 29 on page 436).

If needed, you can manually configure the protocol addresses of other routers using the Frame Relay configuration command **add protocol-address**. For additional information, see the chapter Configuring and Monitoring Frame Relay Interfaces in *8371 Interface Configuration and Software User's Guide*.

Chapter 34. Configuring and Monitoring ARP

This chapter describes how to configure and monitor ARP protocol activity and how to use the ARP monitoring commands. It includes the following sections:

- "Accessing the ARP Configuration Environment"
- "ARP and Inverse ARP Configuration Commands"
- "Accessing the ARP Monitoring Environment" on page 443
- "ARP Monitoring Commands" on page 443

Accessing the ARP Configuration Environment

For information on how to access the ARP configuration environment, see"Getting Started" in *8371 Interface Configuration and Software User's Guide*.

Use the following procedure to access the ARP *configuration* process.

1. At the OPCON prompt, enter **talk 6**. For example:

```
* talk 6
Config>
```

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

2. At the CONFIG prompt, enter the **prot arp** command to get to the `ARP Config>` prompt.

ARP and Inverse ARP Configuration Commands

This section describes the ARP configuration commands. Table 64 lists the ARP configuration commands. You can access ARP configuration commands at the `ARP config>` prompt.

Table 64. ARP Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add Entry	Add a MAC address translation entry.
Change Entry	Change a MAC address translation entry.
Delete Entry	Deletes a MAC address translation entry.
Disable Auto-refresh	Disable ARP auto-refresh.
Enable Auto-refresh	Enable ARP auto-refresh.
List	List ARP configuration data in SRAM.
Set	Set the usage and refreshes timeout values.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

Add Entry

Use the **add entry** command to add a "static protocol-to-hardware address mapping" entry. This command is currently supported for IP addresses only.

© Copyright IBM Corp. 1996, 1999

Syntax:

add entry *ifc# prot-type prot-addr MAC-addr*

ifc# **Valid values**: Any defined interface

Default value: 0

prot-type

Valid values: Any protocol that ARP supports.

Default value: IP

prot-addr

Valid Values: Any valid IP address

Default Value: 0

MAC-addr

Valid Values: Any valid MAC address

Default Value: None

Example: **add entry**

```
Interface Number [0]?
Protocol [IP]?
IP Address [0.0.0.0]?
Mac Address []?
```

Change Entry

Use the **change entry** command to change a “static protocol-to-hardware address mapping” entry. This command is currently supported for IP addresses only. The hardware address parameter (MAC-addr) should be the address of the node being changed.

Syntax:

change entry *ifc# prot-type prot-addr MAC-addr*

ifc# **Valid values**: Any defined interface

Default value: 0

prot-type

Valid values: Any protocol that ARP supports.

Default value: IP

prot-addr

Valid Values: Any valid IP mask

Default Value: None

MAC-addr

Valid Values: Any valid MAC address

Default Value: None

Example: **change entry**

```
Interface Number [0]?
Protocol [IP]?
IP Address [0.0.0.0]?
Mac Address []?
```

Delete Entry

Use the **delete entry** command to delete a “static protocol-to-hardware address mapping” entry. This command is currently supported for IP addresses only.

Syntax:

delete entry *ifc# prot-type prot-addr*

ifc# **Valid values**: Any defined interface

Default value: 0

prot-type

Valid values: *IP*

Default value: IP

prot-addr

Valid Values: Any valid IP address

Default Value: 0.0.0.0

Example: `delete entry`

```
Interface Number [0]?
Protocol [IP]?
IP Address [0.0.0.0]?
```

Disable Auto-Refresh

Use the **disable auto-refresh** command to disable the auto-refresh function. The auto-refresh function is the router’s capability to send an ARP request based on the entry in the translation cache before the refresh timer expires. The request is sent directly to the hardware address in the current translation instead of a broadcast. If auto-refresh is disabled, no ‘preemptive’ ARP request is made, the refresh timer is allowed to expire, and the ARP translation is purged from the table. The next protocol packet to the destination protocol address will then cause a new ARP request to be broadcast on the network.

Syntax:

disable auto-refresh

Example: `disable auto-refresh`

Enable Auto-Refresh

Use the **enable auto-refresh** command to enable the auto-refresh function. The auto-refresh function is the router’s capability to send an ARP request based on the entry in the translation cache before the refresh timer expires. The request is sent directly to the hardware address in the current translation instead of a broadcast.

Enabling auto-refresh could cause entries to be retained in the cache regardless of their usage. On networks with a large number of nodes, this can lead to an excessive number of entries in the cache, which might adversely affect router performance. However, on networks with a small number of nodes, this option is useful in reducing broadcast ARP traffic.

The auto-refresh function is enabled by default.

Syntax:

enable auto-refresh

Example: `enable auto-refresh`

List

Use the **list** command to display the contents of the router's ARP configuration as stored in SRAM. The list command displays the current timeout settings for the refresh and usage timer.

Syntax:

list all
config
entry

all Lists the ARP configuration followed by all of the ARP entries.

Example: `list all`

```
ARP configuration:

Refresh Timeout: 5 minutes
Auto Refresh: disabled

Mac address translation configuration
IF #           Prot #         Protocol --> Mac Address
 0               0          2.2.2.1 --> 0000C90932EF
```

config Lists the configuration for the different ARP parameters.

Example: `list config`

```
ARP configuration:

Refresh Timeout: 5 minutes
Auto refresh: disabled
```

entry Lists the ARP entries in SRAM.

Example: `list entry`

```
Mac address translation configuration

IF #           Prot #         Protocol --> Mac Address
 0               0          2.2.2.1 --> 0000C90932EF
```

Set

Use the **set** command to set an ARP configuration parameter.

Syntax:

set refresh-timer

refresh-timer *minutes*

Changes the timeout value for the refresh timer. To change the timeout value for the refresh timer, enter the timeout value in minutes. A setting of zero (0) turns off (disables) the refresh timer.

This timer is used in determining when an ARP translation cache entry is to be refreshed while auto-refresh is enabled, or purged while auto-refresh is disabled. Disabling the timer causes entries to be retained until a newly learned address translation causes entries to be removed, until entries are cleared manually with the ARP **clear** monitoring command, or until the router is restarted.

Valid Values: An integer number of minutes in the range of 0 to 65535

Default Value: 5 minutes

Example: `set refresh-timer 3`

Accessing the ARP Monitoring Environment

Use the following procedure to access the ARP monitoring commands. This process gives you access to the ARP *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. For example:

```
* talk 5
+
```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the + prompt, enter the **protocol arp** command to get you to the ARP> prompt.

   Example:

```
+ prot arp
ARP>
```

ARP Monitoring Commands

This section describes the ARP monitoring commands. You can access ARP monitoring commands at the ARP> prompt. Table 65 shows the commands.

Table 65. ARP monitoring Commands

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Clear	Clear the cache for a specified interface.
Dump	Display the cache for a specified interface.
Hardware	List each ARP-configured network.
Ping	Verify connectivity between the device and the specified end station.
Protocol	List each ARP-configured protocol.
Statistics	Display ARP information.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Clear

Use the **clear** command to flush the ARP cache for a given network interface. The **clear** command can be used to force the deletion of bad transactions.

To clear a particular interface, enter the interface or network number as part of the command. To obtain the interface number, use the CONFIG **list devices** command.

Syntax:

clear *interface#*

Example: `clear 1`

Dump

Use the **dump** command to display the ARP cache for a given network/protocol combination. To display the ARP cache for a particular interface, enter the interface or network number as part of the command. To obtain the interface number, use the CONFIG **list devices** command.

If there is more than one protocol on that network, the protocol number must also be given. This causes the monitoring to display the hardware address-to-protocol mappings stored in that database. If ARP is in use by only one protocol on the specified interface, then the protocol number is optional. To obtain the protocol number, use the CONFIG **protocol** command.

The **dump** command display shows the hardware address, the protocol address, and the refresh timer parameter for each mapping.

Syntax:

dump *interface# protocol#*

Example: `dump 2 ip`

```
Hardware Address    IP Address      Refresh
02-07-01-00-00-01   192.9.1.2       Permanent
a1-b2-c3-4d-5e-6f   128.185.214.36  5
100                 128.185.123.51  Not Aging
16                  128.185.214.38  Not Aging
```

Valid refresh timer parameters are:

Permanent
A statically configured mapping between hardware address and protocol address (entered using the ARP **add entry** command. These entries do not age and are not overwritten by dynamically learned mappings.

minutes to expire
The number of minutes until this mapping expires due to aging or until this mapping is refreshed (if auto-refresh is enabled). This parameter is expressed as a numeric value.

Not Aging
A fixed SVC or PVC mapping learned through Inverse ARP. It begins to age only when the circuit goes down. The mapping can be overwritten by a newer learned address and can be cleared by the ARP **clear** monitoring command.

Hardware

Use the **hardware** command to display the networks registered with ARP. The **hardware** command lists each ARP-registered network, and displays each network's hardware address space (Hardware AS) and local hardware address.

Syntax:

hardware

Example: `hardware`

```
Network       Hardware AS    Hardware Address

 1 FR/0          000F            1023
 5 TKR/0         0006            00:00:C9:09:32:EF
 8 Eth/0         0001            AA-00-04-00-26-14
 9 IPPN/0        2048            128.185.214.38
10 BDG/0         0001            00-00-93-90-4C-F7
```

Note: The IPPN entry refers to IP Tunneling where the hardware address field indicates the IP address of the IP Tunnel.

Ping

Use the **ping** command to have the router send ICMP Echo Requests to a given destination. For more information on the **ping** command, see “Ping” on page 362.

Protocol

Use the **protocol** command to display (by network) the protocols that have addresses registered with ARP. This command displays the network, protocol name, protocol number, protocol address space (in hexadecimal), and local protocol addresses.

Syntax:

protocol

Example: `protocol`

```
Network  Protocol (num)  AS    Protocol Address(es)
5 TKR/0  IP       (00)   800    128.185.209.38
6 TKR/1  IP       (00)   800    10.1.181.38
8 Eth/0  IP       (00)   800    128.185.221.38
```

Note: SR entries refer to Source Routing - the protocol address is used to indicate the MAC address. Use the token-ring **dump** command to view actual RIF entries.

Statistics

Use the **statistics** command to display a variety of statistics about the operation of the ARP module.

Syntax:

statistics

Example: `statistics`

```
ARP input packet overflows
Net   Count
 PPP/0  0
 PPP/1  0
 TKR/0  0
IPPN/0  0
 BDG/0  0

ARP cache meters
Net Prot  Max Cur Cnt    Alloc  Refresh: Tot  Failure  TMOs: Refresh
0   0     1   1   1       17             0        0               13
0  22     1   0   0        6             0        0                6
1   0     1   1   2       27             0        0               25
1  16     3   3   7      291             0        0                0
```

ARP monitoring Commands (Talk 5)

```
                    2   0     1   0   0        2            0        0              2
                    2  16     1   0   0        1            0        0              0
                    8   0     1   1   1       11            0        0             10
```

ARP input packet overflows	Displays counters that represent the number of ARP packets discarded on input because the ARP layer was too busy. The counts shown are per network interface.
ARP cache meters	Consists of a variety of meters on the operation of the ARP cache. The counts shown are all per protocol, per interface.
Net	Displays the interface numbers.
Prot	Displays the protocol numbers.
Max	Displays the all-time maximum length hash chain.
Cur	Displays the current maximum length hash chain.
Cnt	Displays the count of entries currently active.
Alloc	Displays the count of entries created.
Rfrsh:Tot	Displays the number of refresh requests sent for this network interface and protocol.
Fail	Displays the number of auto-refresh attempt failures due to unavailability of internal resources. This count is not related to whether or not an entry was refreshed.
TMOs:Rfrsh	Displays the count of entries deleted due to a timeout of the refresh timer.

Chapter 35. Using OSPF

This chapter describes how to use the Open Shortest Path First (OSPF) Protocol, which is an Interior Gateway Protocol (IGP). The router supports the following IGPs for building the IP routing table, Open Shortest Path First (OSPF) Protocol and RIP Protocol. OSPF is based on link-state technology or the shortest-path first (SPF) algorithm. RIP is based on the Bellman-Ford or the distance-vector algorithm.

Included in this chapter are the following sections:

Routers that use a common routing protocol form an *autonomous system* (AS). This common routing protocol is called an Interior Gateway Protocol (IGP). IGPs dynamically detect network reachability and routing information within an AS and use this information to build the IP routing table. IGPs can also import external routing information into the AS. The router can simultaneously run OSPF and RIP. When it does, OSPF routes are preferred. In general, use of the OSPF protocol is recommended due to its robustness, responsiveness, and decreased bandwidth requirements.

The OSPF Routing Protocol

The router supports a complete implementation of the OSPF routing protocol, as specified in RFC 1583 (Version 2). OSPF is a link-state dynamic routing protocol that detects and learns the best routes to reachable destinations. OSPF can quickly perceive changes in the topology of an AS, and after a short convergence period, calculate new routes. The OSPF protocol does not encapsulate IP packets, but forwards them based on the destination address only.

OSPF Routing Summary

When a router is initialized, it uses the Hello Protocol to send Hello packets to its neighbors, and they in turn send their packets to the router. On broadcast and point-to-point networks, the router dynamically detects its neighboring routers by sending the Hello packets to the multicast address *ALLSPFRouters (224.0.0.5);* on non-broadcast networks you must configure information to help the router discover its *neighbors*. On all multi-access networks (broadcast and non-broadcast), the Hello Protocol also elects a *designated router* for the network.

Note: If you are using LAN Emulation, the network is treated as a broadcast network, and you should configure OSPF accordingly. If you are using both RFC 1577 and LAN Emulation on a single physical interface, configure OSPF non-broadcast on the RFC 1577 interfaces (IP addresses assigned to the real interface, for example, ATM/0), and configure OSPF broadcast on virtual or emulated interfaces (IP addresses assigned to emulated or virtual interfaces, for example, TKR/0).

© Copyright IBM Corp. 1996, 1999

The router then attempts to form adjacencies with its neighbors to synchronize their topological databases. Adjacencies control the distribution (sending and receiving) of the routing protocol packets as well as the distribution of the topological database updates. On a multi-access network, the designated router determines which routers become adjacent.

A router periodically advertises its status or link state to its adjacencies. *Link state advertisements* (LSAs) flood throughout an area, ensuring that all routers have exactly the same topological database. This database is a collection of the link state advertisements received from each router belonging to an area. From the information in this database, each router can calculate a shortest path tree with itself designated as the root. Then the shortest path tree generates the routing table.

OSPF is designed to provide services that are not available with RIP. OSPF includes the following features:

- *Least-Cost Routing.* Allows you to configure path costs based on any combination of network parameters. For example, bandwidth, delay, and dollar cost.
- *No limitations to the routing metric.* While RIP restricts the routing metric to 16 hops, OSPF has no restriction.
- *Multipath Routing.* Allows you to use multiple paths of equal cost that connect the same points. You can then use these paths for load distribution that results in more efficient use of network bandwidth.
- *Area Routing.* Decreases the resources (memory and network bandwidth) consumed by the protocol and provides an additional level of routing protection.
- *Variable-Length Subnet Masks.* Allows you to break an IP address into variable-size subnets, conserving IP address space.
- *Routing Authentication.* Provides additional routing security.

OSPF supports the following physical network types:

- *Point-to-Point.* Networks that use a communication line to join a single pair of routers. A 56-Kbps serial line that connects two routers is an example of a point-to-point network.
- *Broadcast.* Networks that support more than two attached routers and are capable of addressing a single physical message to all attached routers.
- *Non-Broadcast Multi-Access (NBMA).* Networks that support more than two attached routers but have no broadcast capabilities. An X.25 Public Data Network is an example of a non-broadcast network. For OSPF to function correctly, this network requires extra configuration information about other OSPF routers attached to the non-broadcast network. Classical IP over ATM (RFC 1577) treats the ATM interface as a Non-Broadcast Multiple Access (NBMA) interface.
- *Point-to-Multipoint.* Networks that support more than two attached routers, have no broadcast capabilities, and are non-fully meshed. A Frame Relay network without PVC between all the attached routers is an example of a Point-to-Multipoint network. Like non-broadcast networks, extra configuration information about other OSPF routers attached to the network is required.

Designated Router

Every broadcast or non-broadcast multi-access network has a designated router that performs two main functions for the routing protocol: it originates network link advertisements and it becomes adjacent to all other routers on the network.

When a designated router originates network link advertisements, it lists all the routers, including itself, currently attached to the network. The link ID for this advertisement is the IP interface address of the designated router. By using the subnet/network mask, the designated router obtains the IP network number.

The designated router becomes adjacent to all other routers and is tasked with synchronizing the link state databases on the broadcast network.

The OSPF Hello protocol elects the designated router after determining the router's priority from the *Rtr Pri* field of the Hello packet. When a router's interface first becomes functional, it checks to see if the network currently has a designated router. If it does, it accepts that designated router regardless of that router's priority, otherwise, it declares itself the designated router. If the router declares itself the designated router at the same time that another router does, the router with higher router priority (*Rtr Pri*) becomes the designated router. If both *Rtr Pris* are equal, the one with the higher router ID is elected.

Once the designated router is elected, it becomes the end-point for many adjacencies. On a broadcast network, this optimizes the flooding procedure by allowing the designated route to multicast its Link State Update packets to the address ALLSPFRouters (224.0.0.5) rather than sending separate packets over each adjacency.

Configuring OSPF

The following sections present information on how to initially configure the OSPF protocol. This information outlines the tasks required to get the OSPF protocol up and running. Information on how to make further configuration changes is explained under "OSPF Configuration Commands" on page 463.

The following steps outline the tasks required to get the OSPF protocol up and running. The sections that follow explain each step in detail, including examples.

Before your router can run the OSPF protocol, you must:

1. Enable the OSPF protocol. In doing so, you must estimate the final size of the OSPF routing domain. (See "Enabling the OSPF Protocol" on page 450.)
2. Set the OSPF router ID. For network technologies that do not support data-link multicast or broadcast (for example, Frame Relay), the multicast datagram must be replicated by the router and forwarded as a data-link unicast. (See "Setting OSPF Router IDs" on page 450.)
3. Define OSPF areas attached to the router. If no OSPF areas are defined, a single backbone area is assumed. (See "Defining Backbone and Attached OSPF Areas" on page 450.)
4. Define the router's OSPF network interfaces. Set the cost of sending a packet out on each interface, along with a collection of the OSPF operating parameters. (See "Setting OSPF Interfaces" on page 454.)
5. If the router interfaces to non-broadcast networks set additional interface parameters. (See "Setting Non-Broadcast Network Interface Parameters" on page 456 and "Configuring Wide Area Subnetworks" on page 456.)
6. If you want the router to import routes learned from other routing protocols running on this router (BGP, RIP or statically configured routes), enable AS boundary routing. In addition, you must define whether routes are imported as Type 2 or Type 1 externals. (See "Enabling AS Boundary Routing" on page 458.)

7. If you want to boot via a neighboring router over an attached point-to-point or point-to-multipoint interface, you must configure the neighbor's IP address. Do this by adding an OSPF neighbor for the point-to-point interface's destination.

Enabling the OSPF Protocol

When enabling the OSPF routing protocol, you must supply the following two values to estimate the final size of the OSPF routing domain:

- Total number of AS external routes that will be imported into the OSPF routing domain. A single destination might lead to multiple external routes when it is imported by separate AS boundary routers. For example, if the OSPF routing domain has two AS boundary routers, both importing routes to the same 100 destinations, set the number of AS external routes to 200.
- Total number of OSPF routers in the routing domain.

Configure these two values identically in all of your OSPF routers. Each router running the OSPF protocol has a database describing a map of the routing domain. This database is identical in all participating routers. From this database the IP routing table is built through the construction of a shortest-path tree, with the router itself as root. The routing domain refers to an AS running the OSPF protocol.

To enable the OSPF routing protocol, use the **enable** command as shown in the following example.

```
OSPF Config> enable ospf
Estimated # external routes[100]? 200
Estimated # OSPF routers [50]? 60
Maximum Size LSA [0]? 2048
```

Normally, 2048 bytes is large enough for any link state advertisement (LSA) generated by the router. However, routers with many OSPF dial links (for example, ISDN dial links) can require a larger LSA. Additionally, in these situations, the **packet-size** may also need to be increased in the general configuration.

Setting OSPF Router IDs

Every router in an OSPF routing domain must be assigned a unique 32-bit router ID. Choose the value used for the OSPF router ID as follows:

- If you use the IP configuration **set router ID** command, the value configured is used as an OSPF router ID. The configured OSPF router ID must be one of the router's IP addresses or the internal address.
- If you use the IP configuration **set internal address** command, the address configured is used as the OSPF router ID. It is recommended that the same value be used for the router ID and internal address, if defined.
- If neither the router ID nor the internal address are configured during IP configuration, the first OSPF interface address will be used as the OSPF router ID.

Defining Backbone and Attached OSPF Areas

Figure 30 on page 452 shows a sample diagram of the structure of an OSPF routing domain. One division is between IP subnetworks within the OSPF domain and IP subnetworks external to the OSPF domain. The subnetworks included within the OSPF domain are subdivided into regions called *areas*. OSPF areas are collections of contiguous IP subnetworks. The function of areas is to reduce the OSPF overhead required to find routes to destinations in a different area. Overhead

is reduced both because less information is exchanged between routers and because fewer CPU cycles are required for a less complex route table calculation.

Every OSPF routing domain must have at least a *backbone area*. The backbone is always identified by area number 0.0.0.0. For small OSPF networks, the backbone is the only area required. For larger networks with multiple areas, the backbone provides a core that connects the areas. Unlike other areas, the backbone's subnets can be physically separate. In this case, logical connectivity of the backbone is maintained by configuring *virtual links* between backbone routers across intervening non-backbone transit areas.

Routers that attach to more than one area function as area *border routers*. All area border routers are part of the backbone, so a border router must either attach directly to a backbone IP subnet or be connected to another backbone router over a virtual link. In addition, there must be a collection of backbone subnetworks and virtual links that connects all of the backbone routers.

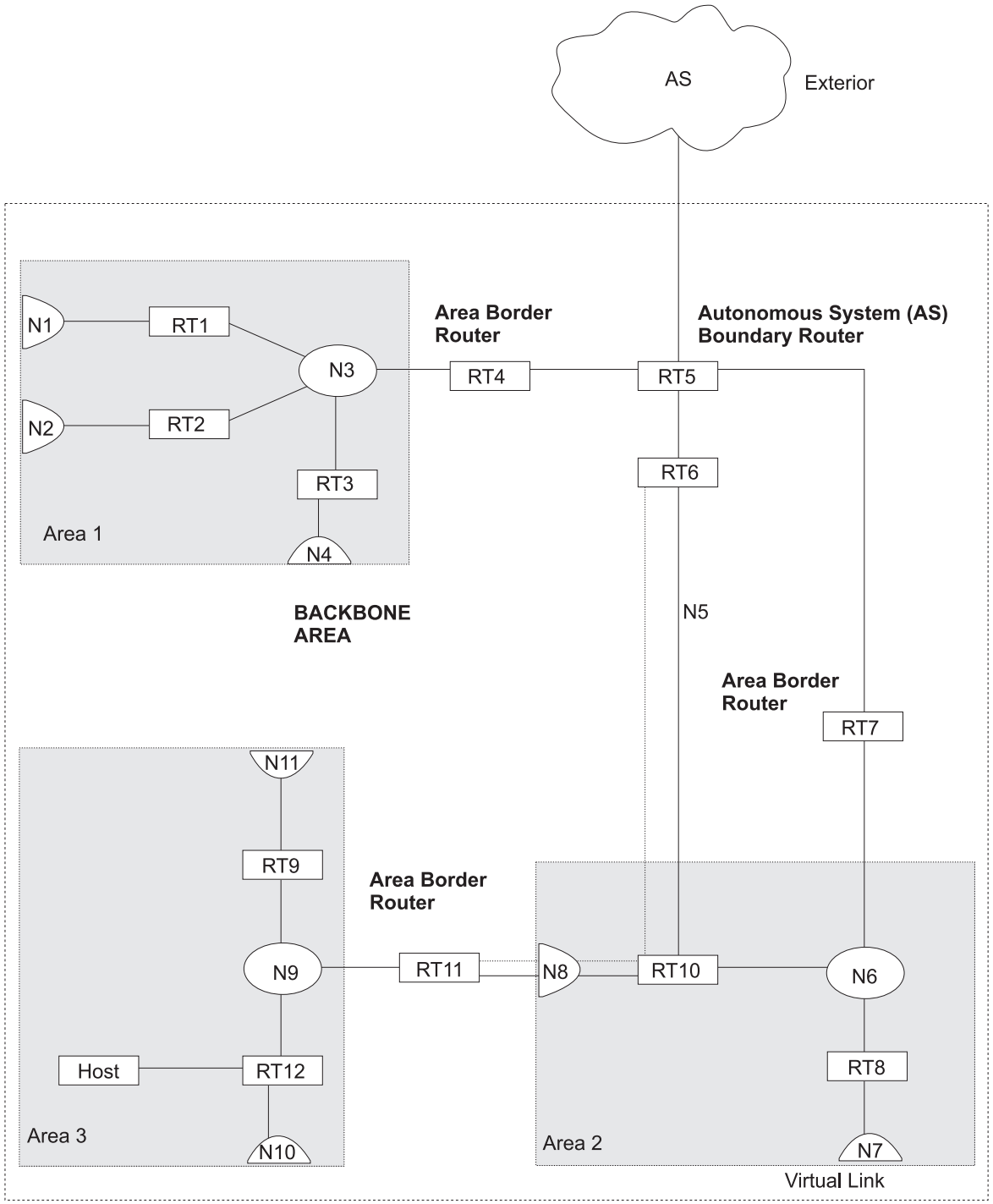

Figure 30. OSPF Areas

The information and algorithms used by OSPF to calculate routes vary according to whether the destination IP subnetwork is within the same area, in a different area within the same domain, or external to the OSPF domain. Every router maintains a complete map of all links within its area. All router to multi-access network, network to multi-access router, and router to router links are included in the map. A shortest path first algorithm is used to calculate the best routes to destinations within the area from this map. Routes between areas are calculated from summary advertisements originated by area border routers for IP subnetworks, IP subnetwork

ranges, and autonomous system external (ASE) boundary routers located in other areas of the OSPF domain. External routes are calculated from ASE advertisements that are originated by ASE boundary routers and flooded throughout the OSPF routing domain.

The backbone is responsible for distributing inter-area routing information. The backbone area consists of any of the following:

- Networks belonging to Area 0.0.0.0
- Routers attached to those networks
- Routers belonging to multiple areas
- Configured virtual links

Defining Attached Areas

A stub area is an area that allows no type 5 LSAs to be propagated into the area and instead depends on default routing to external destinations. A common type of stub area has only one router through which traffic from the stub area exits to the other devices of the network.

OSPF ASE advertisements are never flooded into stub areas. In addition, the **set area** command has an option to suppress origination into the stub of summary advertisements for inter-area routes. A summary advertisement is a type 3 LSA and is used by area border routers to advertise inter-area routes. If you choose to inhibit the advertisement of summary LSAs, then the area border router will advertise a type 3 default route into the stub. As a result, traffic within the stub destined for unknown IP subnets is forwarded to the area border router over the default route. The border router uses its more complete routing information to forward the traffic on an appropriate path toward its destination.

You can define an area as a stub when:

1. There is no requirement for the area to handle transit backbone traffic.
2. It is acceptable for area routers to use an area-border-router-generated default for traffic destined outside the AS.
3. There is no requirement for area routers to be AS boundary routers (OSPF routers that advertise routes from external sources as AS external advertisements). An external source is a network running a protocol other than OSPF.

In this case, only the area border routers and backbone routers will have to calculate and maintain AS external routes.

An area cannot be configured as a stub if it is used as a transit area for virtual links.

To set the parameters for an OSPF stub area, use the **set area** command and respond to the following prompts:

```
Area number [0.0.0.0]? 0.0.0.1
Is this a stub area? [No]:  yes
Stub default cost? [0]:
Import summaries? [Yes]:
```

Defining Subnet Address Ranges for an Attached Area

The propagation of LSAs is also decreased by defining subnet address ranges for an area attached to an area border router. A range is defined by an IP address and an address mask. Subnets are considered to fall within the range if the subnet IP address and the range IP address match after the range mask has been applied to both addresses.

When a range is added, the area border router suppresses summary advertisements for subnets in the areas that are included in the range. The suppressed advertisements would have been originated into the other areas to which the border router is attached. Instead, the area border router may originate a single summary advertisement for the range or no advertisement at all, depending upon the option chosen with the **add range** command.

Note that if the range is not advertised, there will be no inter-area routes for any destination that falls within the range. Also note that ranges cannot be used for areas that are used as transit areas by virtual links.

Setting OSPF Interfaces

OSPF interfaces are a subset of the IP interfaces defined during IP configuration. The parameters configured for OSPF interfaces determine the topology of the OSPF domain, the routes that will be chosen through the domain, and the characteristics of the interaction between directly connected OSPF routers. The **set interface** command is used to define an OSPF interface and to specify some of its characteristics. Other characteristics of the interface were specified in response to the **add address** prompt during IP configuration.

OSPF Domain Topology

The definition of the topology of an OSPF domain depends on a definition of which routers are directly connected across some physical media or subnetwork technology and the area to which those connections are a part. The basic case is for all routers attached to a physical subnetwork to be directly connected, but it is possible to define multiple IP subnetworks over a single physical subnetwork. In that case, OSPF will consider routers to be directly connected only when they have OSPF interfaces attached to the same IP subnetwork. It is also possible to have cases where routers attached to the same subnetwork do not have a direct link layer connection.

For LAN media, directly connected OSPF routers are determined from the IP subnetwork and physical media associated with an OSPF interface. The IP address of the OSPF interface is specified in response to the **Interface IP address** prompt. This address must match the address of an IP interface that was defined with the **add address** command during IP configuration. The IP address, along with the subnetwork mask defined with the **add address** command determine the IP subnetwork to which the OSPF interface attaches. The *net index* associated with the IP interface by the add address command determines the physical subnetwork to which the OSPF interface attaches. The broadcast capability of LANs allows OSPF to use multicast Hello messages to discover other routers that have interfaces attached to the same IP subnetwork. Consequently, the interface parameters are all that are required for OSPF to determine which routers are directly connected across a LAN.

LANs can be used to connect an OSPF router with IP hosts. In this case, it is still necessary to define an OSPF interface to any IP subnetwork that is defined for the LAN. Otherwise, OSPF will not generate routes with those IP subnetworks as destinations. To prevent OSPF Hello traffic on these LANs without other attached routers, the network can be defined as a non-broadcast multi-access network. The router priority should also be set to zero because no designated router is required.

The requirements for configuring OSPF interfaces that attach to serial lines vary with the lower layer technology.

For point-to-point lines, only one other router is accessible over the interface, so the directly connected router can be determined without additional configuration. In fact, because there is no requirement to configure an IP subnetwork at all, unnumbered OSPF interfaces can be used for point-to-point lines. In this case, the same net index used as the IP address for the IP add address command is used as the IP address for the OSPF set interface command.

For subnetwork technologies like Frame Relay, ATM, and X.25 that support connections to multiple routers over a single serial line, the configuration of the OSPF interfaces is similar to that for a LAN, but because directly connected routers are not discovered dynamically for these subnetwork technologies, additional configuration is required to specify directly connected neighbors. For more information on the required configuration, see “Configuring Wide Area Subnetworks” on page 456.

Costs for OSPF Links

OSPF calculates routes by finding the least-cost path to a destination. The cost of each path is the sum of the costs for the different links in the path. The cost of a link to a directly connected router is specified at the **set interface** command for **Type of Service 0 cost**.

Correctly configuring the costs according to the desirability of using interfaces for data traffic is critical for obtaining the desired routes through an OSPF domain. The factors that make individual links more or less desirable may vary in different networks, but the most common goal is to choose routes with the least delay and the most capacity. In general, this policy can be achieved by making the cost of a link inversely proportional to the bandwidth of the media used for the physical subnetwork.

A recommended approach is to use a cost of one for the highest bandwidth technology. For example, use the value 1 as the cost for an interface running 155 Mbps ATM.

Table 66. Sample Costs for OSPF Links

Interface Bandwidth	Cost
155 Mbps ATM	1
100 Mbps Ethernet	1
Ethernet	10
16 Mbps Token-Ring	6
4 Mbps Token-Ring	25
serial line	Cost based on bandwidth
Emulated Token-Ring (See note.)	1
Emulated Ethernet (See note.)	1

Note: Ethernet will run at the interface speed (for example, 155 Mbps), and should be configured with a cost of 1.

ATM can for attach to networks at a slower rate than the maximum line speed. For example, if the router has a port that is capable of 155 Mbps, and a router connects to it with 25 Mbps, that link will still be treated as a cost of 1. The OSPF weighting is on an interface basis.

The cost of an OSPF interface can be dynamically changed from the router’s monitoring environment. This new cost is flooded quickly throughout the OSPF routing domain, and modifies the routing immediately.

When the router restarts/reloads, the cost of the interface reverts to the value that has been configured in SRAM.

Interactions Between Neighbor Routers

A number of the values configured with the **set interface** command are used to specify parameters that control the interaction of directly connected routers. They include:

- Retransmission interval
- Transmission delay
- Router priority
- Hello interval
- Dead router interval
- Demand Circuit
- Hello Suppression
- Poll Interval
- Authentication key

In most cases, the default values can be used.

Note: The Hello interval, the dead router interval, and the authentication key must have the same value for all OSPF routers that attach to the same IP subnetwork. If the values are not the same, routers will fail to form direct connections (adjacencies).

Setting Non-Broadcast Network Interface Parameters

If the router is connected to a non-broadcast, multi-access network, you have to configure the following parameters to help the router discover its OSPF neighbors. This configuration is necessary only if the router will be eligible to become designated router of the non-broadcast network.

First configure the OSPF poll interval with the following command:

```
OSPF Config> set non-broadcast
Interface IP address [0.0.0.0]? 128.185.138.19
Poll Interval [120]?
```

Then configure the IP addresses of all other OSPF routers that will be attached to the non-broadcast network. For each router configured, you must also specify its eligibility to become the designated router.

```
OSPF Config> add neighbor
Interface IP address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
Can that router become Designated Router [Yes]?
```

Setting non-broadcast can also be used to force a network without any other OSPF routers to be advertised. The router priority for the interface should be set to zero and no neighbors should be defined.

Configuring Wide Area Subnetworks

Frame Relay and X.25 allow direct connections between multiple routers over a single serial line. Additional configuration beyond that achieved with the **set interface** command is required for OSPF interfaces that attach to this kind of network. Because OSPF protocol messages are sent directly to specific neighbors on these networks, configuration is used instead of dynamic discovery to determine neighbor relationships and router roles.

Note: The configurations described in this section do not apply to point-to-point networks.

OSPF can assume either of two patterns for the direct connections between routers across these subnetworks:

- Point-to-Multipoint
- Non-broadcast multi-access (NBMA)

The key factor that distinguishes these two patterns is whether or not there is a direct connection between all pairs of routers that attach to the subnetwork (*full mesh connectivity*) or whether some of the routers are only connected through multi-hop paths with other routers as intermediates (*partial mesh connectivity*).

Non-broadcast multi-access (NBMA) requires *full mesh connectivity* while point-to-multipoint requires only *partial mesh connectivity*.

Point-to-multipoint is the default choice because it works for both full mesh connectivity and partial mesh connectivity. But when full mesh connectivity is available, NBMA is a more efficient solution.

Configuring Point-to-Multipoint Subnetworks

Point-to-multipoint can be configured more easily than NBMA because there are no DRs, but neighbor relationships must be configured for all pairs of routers that will exchange data traffic directly across the point-to-multipoint subnet. Each pair of directly connected routers will exchange Hello messages, so one side can discover the other through these messages. The router configured to send the first Hello message, however, must have the IP address of its neighbor configured using the **add neighbor** command.

It is important to remember that OSPF will not calculate the correct routes if some of the routers attached to a subnetwork represent it as NBMA and others represent it as point-to-multipoint. Therefore, never use the **set non-broadcast** command for any interface to a point-to-multipoint network.

Configuring NBMA Subnetworks

For NBMA IP subnetworks, some subset of the attached OSPF routers are configured to be eligible to be the designated router (DR). Each router eligible to be the DR periodically sends Hello messages to all other routers eligible to be the DR. These messages are used in the protocol to elect a DR and a backup DR. Both the DR and the backup DR periodically exchange Hello messages with all other OSPF routers that are attached to the NBMA IP subnetwork. Also, the flow of OSPF route information across the NBMA IP subnetwork is only between each of the attached routers and the DR or backup DR.

Select NBMA by using the **set non-broadcast** command for interfaces that attach to an NBMA subnetwork. This command must be used for all interfaces that attach to the NBMA network.

The configuration required for an OSPF router that attaches to an NBMA subnetwork depends on whether or not that router is eligible to become the DR.

- For a router not eligible to become a DR, the **set interface** command must be used to set the router priority to 0.
- For a router eligible to become a DR, the **set interface** command must be used to set the router priority to a nonzero value and the **add neighbor** command

must be used to identify all of the OSPF routers with interfaces attached to the NBMA subnetwork and to indicate which of them are eligible to become DR.

Note: In a star configuration, use the **add neighbor** command at the hub (neighbors at the remote site do not need to be configured). The **add neighbor** command takes effect immediately without restarting the router.

Enabling AS Boundary Routing

To import routes learned from other protocols (RIP and statically configured information) into the OSPF domain, enable AS boundary routing. You must do this even if the only route you want to import is the default route (destination 0.0.0.0).

When enabling AS boundary routing, you are asked which external routes you want to import. You can choose to import, or not to import, routes belonging to the following categories.

- BGP routes
- RIP routes
- Static routes
- Direct routes

For example, you could choose to import BGP and direct routes, but not RIP or static routes.

Independently of the above external categories, you can also configure whether or not to import subnet routes into the OSPF domain. This configuration item defaults to ENABLED (subnets are imported).

The metric type used in importing routes determines how the imported cost is viewed by the OSPF domain. When comparing two type 2 metrics, only the external cost is considered in picking the best route. When comparing two type 1 metrics, the external and internal costs of the route are combined before making the comparison. For example, you can set the router so that its default is originated only if a route to 10.0.0.0 is received from AS number 12. Setting the AS number to 0 means “from any AS.” Setting the network number to 0.0.0.0 means “any routes received.”

The syntax of the **enable** command is as follows:

```
OSPF Config>enable as boundary
Import BGP routes? [No]: yes
Import RIP routes? [No]:
Import static routes? [No]:
Import direct routes? [No]: yes
Import subnet routes? [No]:
Always originate default route? [No]: yes
Originate as type 1 or 2 [2]? 2
Default route cost [1]?
Default forwarding address [0.0.0.0]? 10.1.1.1
```

See the command **enable as boundary routing** on page 467 for information about using a route filter policy to define AS boundary routing parameters.

Configuring OSPF over ATM

The options for configuring OSPF over an ATM subnetwork depend on whether LAN Emulation or Classical IP over ATM is being used for the IP layer. In the case of LAN Emulation, OSPF is configured in the same way as for a real LAN. For Classical IP over ATM the OSPF configuration options are the same as for Wide

Area Subnetworks. See “Configuring Wide Area Subnetworks” on page 456. Both NBMA and point-to-multipoint configurations are supported.

Other Configuration Tasks

Setting Virtual Links

To maintain backbone connectivity, you must have all of your backbone routers interconnected either by permanent or virtual links. You can configure virtual links between any two area border routers that share a common non-backbone and non-stub area. Virtual links are considered to be separate router interfaces connecting to the backbone area. Therefore, you are asked to also specify many of the interface parameters when configuring a virtual link.

The following example illustrates the configuration of a virtual link. Virtual links must be configured in each of the link’s two end-points. Note that you must enter OSPF router IDs in the same form as IP addresses.

```
OSPF Config>set virtual
Virtual endpoint (Router ID) [0.0.0.0]? 128.185.138.21
Link's transit area [0.0.0.1]?
Retransmission Interval (in seconds) [10]?
Transmission Delay (in seconds) [5]?
Hello Interval (in seconds) [30]?
Dead Router Interval (in seconds) [180]?
Authentication Type (0 - None, 1 - Simple) [0]? 1
Authentication Key []? 41434545
Retype Auth. Key []? 41434545
```

No cost is configured for a virtual link because the cost is the OSPF intra-area cost between the virtual link end-points through the transit area.

Configuring for Routing Protocol Comparisons

If you use a routing protocol in addition to OSPF, or when you change your routing protocol to OSPF, you must set the Routing Protocol Comparison.

OSPF routing in an AS occurs on these three levels: intra-area, inter-area, and exterior.

Intra-area routing occurs when a packet’s source and destination address reside in the same area. Information that is about other areas does not affect this type of routing.

Inter-area routing occurs when the packet’s source and destination addresses reside in different areas of the same AS. OSPF does inter-area routing by dividing the path into three contiguous pieces: an intra-area path from source to an area border router; a backbone path between the source and destination areas; and then another intra-area path to the destination. You can visualize this high-level of routing as a star topology with the backbone as hub and each of the areas as a spoke.

Exterior routes are paths to networks that lie outside the AS. These routes originate either from routing protocols, such as Border Gateway Protocol (BGP), or from static routes entered by the network administrator. The exterior routing information provided by BGP does not interfere with the internal routing information provided by the OSPF protocol.

AS boundary routers can import exterior routes into the OSPF routing domain. OSPF represents these routes as AS external link advertisements.

OSPF imports external routes in separate levels. The first level, called type 1 routes, is used when the external metric is comparable to the OSPF metric (for example, they might both use delay in milliseconds). The second level, called external type 2 routes, assumes that the external cost is greater than the cost of any internal OSPF (link-state) path.

Imported external routes are tagged with 32 bits of information. In a router, this 32-bit field indicates the AS number from which the route was received. This enables more intelligent behavior when determining whether to re-advertise the external information to other autonomous systems.

OSPF has a 4-level routing hierarchy (see Figure 31). The **set comparison** command tells the router where the BGP/RIP/static routes fit in the OSPF hierarchy. The two lower levels consist of the OSPF internal routes. OSPF intra-area and inter-area routes take precedence over information obtained from any other sources, all of which are located on a single level.

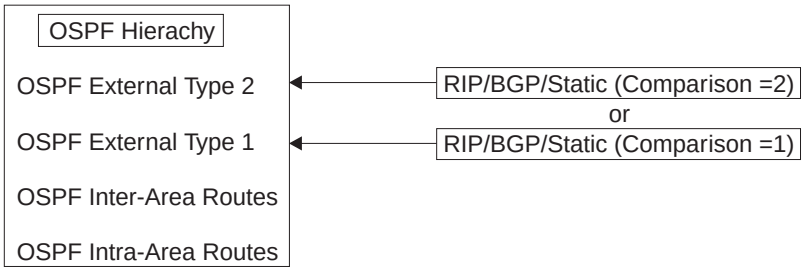

Figure 31. OSPF Routing Hierarchy

To put the BGP/RIP/static routes on the same level as OSPF external type 1 routes, set the comparison to 1. To put the BGP/RIP/static routes on the same level as OSPF external type 2 routes, set the comparison to 2. The default setting is 2.

For example, suppose the comparison is set to 2. In this case, when RIP routes are imported into the OSPF domain, they will be imported as type 2 externals. All OSPF external type 1 routes override received RIP routes, regardless of metric. However, if the RIP routes have a smaller cost, the RIP routes override OSPF external type 2 routes. The comparison values for all of your OSPF routers must match. If the comparison values set for the routers are inconsistent, your routing will not function correctly.

The syntax of the **set comparison** command is as follows:

```
OSPF Config> set comparison
Compare to type 1 or 2 externals [2]?
```

Demand Circuit

A demand circuit can be configured for any interface. There is no dependence on physical media or the model used by OSPF for the route calculation. When the demand circuit is configured and there are no compatibility problems:

- Only Link State Advertisements (LSAs) with real changes will be advertised over the interface. Normally, OSPF's reliable flooding algorithm causes LSAs to be refreshed with a new instance every 30 minutes even if topology changes have occurred.
- The DoNotAge bit will be set for LSAs flooded over the interface. This is required since they will not be refreshed over the interface.

Request Hello Suppression

This is an additional parameter that you can use to configure an interface to request Hello suppression. This parameter will have value for point-to-point and point-to-multipoint interfaces. In addition, the subnetwork the interface attaches to must be able to notify OSPF that data cannot be delivered over a connection. Currently, ATM and ISDN dial-on-demand interfaces are the only interface types on which Hello suppression is supported.

Poll Interval

When Hello suppression is not active, the poll interval is used only with non-broadcast multi-access subnetworks and is set with the **set non-broadcast** command. You can configure this parameter after an interface has been configured as a demand circuit and Hello suppression has been requested. This parameter will be used by OSPF to try to reestablish a connection when a point-to-point line is down because there was a failure to transmit data but the network still appears to be operational.

Converting from RIP to OSPF

To convert your Autonomous System from RIP to OSPF, install OSPF one router at a time, leaving RIP running. Gradually, all your internal routes will shift from being learned via RIP to being learned by OSPF (OSPF routes have precedence over RIP routes). If you want to have your routes look exactly as they did under RIP (in order to check that the conversion is working correctly) use hop count as your OSPF metric. Do this by setting the cost of each OSPF interface to 1.

Remember that the size of your OSPF system must be estimated when the protocol is enabled. This size estimate should reflect the final size of the OSPF routing domain.

After installing OSPF on your routers, turn on AS boundary routing in all those routers that still need to learn routes via other protocols (BGP, RIP, and statically configured routes). The number of these AS boundary routers should be kept to a minimum.

Finally, you can disable the receiving of RIP information on all those routers that are not AS boundary routers.

Dynamically Changing Interface Costs

The cost of an OSPF interface can be dynamically changed from the router's console interface. This new cost flooded quickly throughout the OSPF routing domain, and modifies the routing immediately.

When the router restarts/reloads, the cost of the interface reverts to the value that has been configured in SRAM.

Dynamically Changing OSPF Configuration Parameters

OSPF configuration parameters can be changed dynamically by updating the configuration through the OSPF configuration facility and subsequently resetting the OSPF protocol through the OSPF console. OSPF neighbors, interfaces, areas, and AS boundary routing policy can be added, deleted, or changed using this technique.

In most cases, these changes are completely non-disruptive. For example, adding an OSPF interface will not effect other OSPF interfaces (other than the origination of new OSPF link state advertisements).

Changes that require all of a router's OSPF advertisements to be re-originated will cause OSPF to be restarted. These include:

- Enabling/Disabling Demand Circuits (RFC 1793)
- Changing the value of the router's Router-ID

In most cases, this will be transparent to the users as the only outage will be the time for OSPF neighbor adjacencies to be reestablished.

Since router memory is reserved for OSPF prior to allocating input/output buffers, OSPF cannot be enabled dynamically unless it was enabled at the time of the last router restart. Additionally, the amount of memory reserved for OSPF cannot be increased without a system restart. The amount of memory reserved is determined by the estimates for routers and AS external routes specified in the **enable OSPF** command.

Example:

```
  OSPF Config>enable OSPF
Estimated # external routes [100]? 300
Estimated # OSPF routers [50]? 100
Maximum Size LSA [2048]?
```

Chapter 36. Configuring and Monitoring OSPF

This chapter describes how to configure the Open Shortest Path First (OSPF) Protocol. OSPF is an Interior Gateway Protocol (IGP). The router supports the following IGPs for building the IP routing table, Open Shortest Path First (OSPF) Protocol and RIP Protocol. OSPF is based on link-state technology or the shortest-path first (SPF) algorithm. RIP is based on the Bellman-Ford or the distance-vector algorithm. This chapter includes the following sections:

Accessing the OSPF Configuration Environment

To access the OSPF configuration environment, enter the following command at the `Config>` prompt:

```
Config> protocol ospf
Open SPF-based Routing Protocol configuration monitoring
OSPF Config>
```

OSPF Configuration Commands

Before you can use OSPF, you must configure it using the OSPF configuration commands. The following section summarizes and then explains the OSPF commands.

Note: Except for the commands noted at "Dynamically Changing OSPF Configuration Parameters" on page 461, which cause OSPF to restart immediately with the changed parameters, the OSPF configuration commands are not effective immediately. They remain pending until you issue the Talk 5 **reset ospf** command.

Enter these commands at the `OSPF config>` prompt. Table 67 shows the commands.

Table 67. OSPF Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds to already existent OSPF information. You can add ranges to areas, and neighbors to non-broadcast networks.
Delete	Deletes OSPF information from SRAM.
Disable	Disables the entire OSPF protocol, AS boundary routing capability, demand circuit capability.
Enable	Enables the entire OSPF protocol, AS boundary routing capability, demand circuit capability.
List	Displays OSPF configuration.
Set	Establishes or changes the configuration information concerning OSPF areas, interfaces, non-broadcast networks, or virtual links. This command also allows you to set the way in which OSPF routes are compared with information gained from other routing protocols.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

© Copyright IBM Corp. 1996, 1999

Response to OSPF Configuration Commands

Except for the commands noted at “Dynamically Changing OSPF Configuration Parameters” on page 461, which cause OSPF to restart immediately with the changed parameters, the OSPF configuration (Talk 6) commands do not become effective immediately. They remain pending until you issue the Talk 5 **reset ospf** command.

Add

Use the **add** command to add more information to already existing OSPF information. With this command you can add ranges to areas as well as neighbors to non-broadcast networks.

Syntax:

range . . .
neighbor . .

range *area# IP-address IP-address-mask*

Adds ranges to OSPF areas. OSPF areas can be defined in terms of address ranges. External to the area, a single route is advertised for each address range. For example, if an OSPF area were to consist of all subnets of the class B network 128.185.0.0, it would be defined as consisting of a single address range. The address range would be specified as an address of 128.185.0.0 together with a mask of 255.255.0.0. Outside of the area, the entire subnetted network would be advertised as a single route to network 128.185.0.0.

Ranges can be defined to control which routes are advertised externally to an area. There are two choices:

- When OSPF is configured to advertise the range, a single inter-area route is advertised for the range if at least one component route of the range is active within the area.
- When OSPF is configured not to advertise the range, no inter-area routes are advertised for routes that fall within the range.

Ranges cannot be used for areas that serve as transit areas for virtual links. Also, when ranges are defined for an area, OSPF will not function correctly if the area is partitioned but is connected by the backbone.

Example:

```
add range 0.0.0.2 128.185.0.0 255.255.0.0

inhibit advertisement ? [No]
```

1. The *area number* has:
   Valid Values: Any valid area number
   Default Value: none
2. The *IP address* has:
   Valid Values: Any valid IP address.
   Default Value: none
3. The *IP address mask* has:
   Valid Values: Any valid IP address mask.

Default Value: none

neighbor

Configures neighbors adjacent to the router over this interface. In non-broadcast multi-access networks, neighbors need to be configured only on those routers that are eligible to become the designated router. If no cost is configured, the interface cost is used.

Example: `add neighbor`

```
Interface IP address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
Can that router become Designated Router on this net [Yes]?
Alternate TOS 0 cost [0]? 100
```

1. The *Interface IP address* has:

   Valid Values: Any valid IP address.

   Default Value: None
2. The *IP Address of Neighbor* has:

   Valid Values: Any valid IP address

   Default Value: None
3. Answer the question, `Can that router become designated router on this net?` For point-to-multipoint interfaces, this parameter is not applicable and should be set to “No”.

   Valid Values: Yes or No

   Default Value: Yes
4. `Alternate TOS 0 cost` allows an alternate cost to be used.

   Valid Values: 0 - 65534

   Default Value: 0 (indicates that interface cost should be used).

Delete

Use the delete command to delete OSPF information from SRAM.

Syntax:

delete range . . .
area . . .
interface . . .
neighbor . . .
non-broadcast . . .
virtual-link

range *area# IP-address*

Deletes ranges from OSPF areas.

Example: `delete range 0.0.0.2 128.185.0.0 255.255.0.0`

1. The *area number* of the range has:

   Valid Values: Any valid area address

   Default Value: none
2. The *IP Address of Range* has:

   Valid Values: Any valid IP address.

   Default Value: none
3. The *IP Address Mask of Range* has:

   Valid Values: Any valid IP address mask.

   Default Value: none

OSPF Configuration Commands (Talk 6)

area *area#*

Deletes OSPF areas from the current OSPF configuration.

Example: `delete area 0.0.0.1`

The *area number* has:

Valid Values: Any valid area number.

Default Value: none

interface *interface-IP-address*

Deletes an interface from the current OSPF configuration.

Example: `delete interface 128.185.138.19`

The *interface IP address* has:

Valid Values: Any valid IP address.

Default Value: none

neighbor *interface-IP-address neighbor-IP-address*

Deletes configured neighbors from the current OSPF configuration.

Example: `delete neighbor`

```
Interface IP address [0.0.0.0]? 128.185.138.19
IP Address of Neighbor [0.0.0.0]? 128.185.138.21
```

1. The *interface IP address* has:

   Valid Values: Any valid IP address.

   Default Value: none

2. The *neighbor IP address* has:

   Valid Values: Any valid IP address.

   Default Value: none

non-broadcast *interface-IP-address*

Deletes non-broadcast network information from the current OSPF configuration.

Example: `delete non-broadcast 128.185.133.21`

1. The *interface IP address* has:

   Valid Values: Any valid IP address.

   Default Value: none

virtual-link

Deletes a virtual link that you have set using the **set virtual-link** command.

Example: `delete virtual-link`

```
Virtual endpoint (Router ID) [0.0.0.0]? 10.1.1.1
Link's transit area [0.0.0.1]? 0.0.0.2
```

1. The *virtual endpoint (router ID)* that defines the ID of the virtual neighbor has:

   Valid Values: Any valid IP address.

   Default Value: none

2. The *link's transit area* has:

   Valid Values: Any valid area address.

   Default Value: 0.0.0.1

Disable

Use the **disable** command to disable either the entire OSPF protocol or just the AS boundary routing capability.

Syntax:

disable	as boundary routing
	demand-circuits
	ospf routing protocol
	rfc1583compatibility
	subnet

as boundary routing

Disables the AS boundary routing capability. When disabled, the router will not import external information into the OSPF domain.

Example: `disable as boundary routing`

demand-circuits

Disables the demand circuit capability. When disabled, the router will not indicate that it supports demand circuit processing in its router link's Link State Advertisement (LSA) and will not originate any LSAs with the DoNotAge bit set. If one router in the routing domain or OSPF stub area does not support demand circuits, none of the routers in the routing domain or OSPF stub area will originate DoNotAge LSAs.

Example: `disable demand-circuits`

OSPF routing protocol

Disables the entire OSPF protocol.

Example: `disable OSPF routing protocol`

RFC1583 Compatibility

Disables the AS External route selection that is compatible with RFC 1583. It is recommended that you do not disable RFC1583 compatibility unless you have the same external route accessible through more than one OSPF area and you are experiencing routing loop problems similar to those described in RFC2178. The default is enabled.

Example: `disable rfc1583Compatibility`

subnet

For an interface to a point-to-point serial line, this option disables the advertisement of a stub route to the subnet that represents the serial line rather than the host route for the other router's address. You must supply this router's address for the interface to identify it.

Example:

```
OSPF Config> disable subnet
Interface IP address [0.0.0.0]? 8.24.3.1
```

The *interface IP address* has:

Valid Values: Any valid IP address.

Default Value: none

Enable

Use the **enable** command to enable the OSPF protocol or particular aspects of that protocol, such as the advertisement of a stub to route to a subnet or the AS boundary routing capability.

Syntax:

enable	as boundary routing
	demand-circuits
	least-cost-ranges

ospf routing protocol
rfc1583compatibility
send outage-only
subnet

as boundary routing

Enables the AS boundary routing capability which allows you to import routes learned from other protocols, for example, BGP, RIP, and statically configured information, into the OSPF domain. For additional information on the use of the **enable** command, see “Configuring OSPF” on page 449.

Example:

Example 2:

```
 enable as boundary routing
Use route policy? [No]: Yes
Router Policy Identifier [1-15 characters] [ ]? ospf-import
Always originate default route? [No]:
```

1. The *Use route policy* question indicates whether a configured route policy is used to determine which non-OSPF routes are imported into OSPF as OSPF external routes. If this question is answered **yes**, many of the questions are no longer displayed because they are not applicable when routing policy is configured. Routing policy provides more granularity by specifying which routes are imported.

   Valid Values: yes or no

   Default Value: no

2. The *Router Policy Identifier* question asks for the character string that identifies a configured route filter policy.

   Valid Values: a 1 to 15-character ASCII string

   Default Value: none

3. The *Import BGP* question indicates whether the BGP routes will be imported into OSPF as OSPF external routes.

   Valid Values: Yes or No

   Default Value: No

4. The *Import RIP* question indicates whether the RIP routes will be imported into OSPF as OSPF external routes.

   Valid Values: Yes or No

   Default Value: No

5. The *Import static* question indicates whether the static routes will be imported into OSPF as OSPF external routes.

   Valid Values: Yes or No

   Default Value: No

6. The *Import direct* question indicates whether the direct routes will be imported into OSPF as OSPF external routes.

   Valid Values: Yes or No

   Default Value: No

7. The *Import subnet* question indicates whether the subnet routes will be imported into OSPF as OSPF external routes.

   Valid Values: Yes or No

   Default Value: Yes

8. The *Always originate default route* question indicates whether to unconditionally originate a default route in the form of an OSPF external advertisement.

Valid Values: Yes or No

Default Value: No

9. The *Originate as type 1 or 2* question indicates whether the OSPF-originated default will have an AS external metric type of 1 or 2. Type 1 metrics are considered in the same context as OSPF costs while type 2 metrics are considered higher than any OSPF metric.

   Valid Values: 1 or 2

   Default Value: 2

10. The *Default route cost* is the parameter that specifies the cost that OSPF associates with the default route to its area border router. The cost is used to determine the shortest path for the default route to its area border router.

    Valid Values: 0 to 16777215

    Default Value: 1

11. The *Default forwarding address* is the parameter that specifies the forwarding address that will be used in the imported default route.

    Valid Values: a valid IP address

    Default Value: none

demand-circuits

Enables demand circuit processing for the router. The router will indicate that it supports demand circuit processing in its router link's Link State Advertisement (LSA). The default is enabled so that demand circuits can be deployed without reconfiguring every router in the OSPF routing domain.

```
OSPF Config> enable demand-circuits
```

OSPF routing protocol

Enables the entire OSPF protocol. When enabling the OSPF routing protocol, you must supply the following two values that will be used to estimate the size of the OSPF link state database:

- Total number of AS external routes that will be imported into the OSPF routing domain. A single destination may lead to multiple external routes when it is imported by separate AS boundary routers. For example, if the OSPF routing domain has two AS boundary routers, both importing routes to the same 100 destinations, the number of AS external routes should be set to 200.

  Valid Values: 0 to 65535

  Default Value: 100

- Total number of OSPF routers in the routing domain.

  Valid Values: 0 to 65535

  Default Value: 50

- Additionally, you can specify the maximum LSA size. This value may need to be increased if you have a large router with many OSPF dial links (for example, ISDN primary) in the same OSPF area. Normally, 2048 is more than enough space for any single LSA.

  Valid Values: 2048 to 65535

  Default Value: 2048

  Example: enable OSPF routing protocol

```
Estimated # external routes[100]? 200
Estimated # OSPF routers [50]? 60
Maximum LSA Size [2048]?
```

RFC1583Compatibility

Enables the AS External route selection that is compatible with RFC 1583. The default is enabled.

Example: `enable rfc1583Compatibility`

subnet

For an interface to a point-to-point serial line, this option enables the advertisement of a stub route to the subnet that represents the serial line rather than the host route for the other router's address. You must supply this router's address for the interface to identify it.

Example:

```
OSPF Config> enable subnet
Interface IP address [0.0.0.0]? 8.24.3.1
```

The *interface IP address* has:

Valid Values: Any valid IP address.

Default Value: none

List

Use the **list** command to display OSPF configuration information.

Syntax:

list all | areas | interfaces | neighbors | non-broadcast | virtual-links

all Lists all OSPF-related configuration information.

Example: `list all`

```
                --Global configuration--
          OSPF Protocol:          Enabled
          # AS ext. routes:       300
          Estimated # routers:    100
          Maximum LSA Size:       2048
          External comparison:    Type 2
          RFC 1583 compatibility: Disabled
          AS boundary capability: Enabled
          Import external routes: BGP RIP STA DIR SUB
          Orig. default route:    No (0,0.0.0.0)
          Default route cost:     (1, Type 2)
          Default forward. addr.: 0.0.0.0
                    Inter-area multicast:   Enabled
          Demand Circuits:        Enabled
          Least Cost Ranges:      Disabled
          LSA Max Random Initial Age:    0

                --Area configuration--
     Area ID          AuType          Stub? Default-cost Import-summaries?
     0.0.0.0          0=None          No         N/A           N/A


        --Interface configuration--
     IP address        Area    Cost  Rtrns  TrnsDly  Pri  Hello Dead
     128.185.184.11    0.0.0.1   1      5       1     1    10    60
     128.185.177.11    0.0.0.1   1      5       1     1    10    60
     128.185.142.11    0.0.0.0   1      5       1     1    10    60
```

OSPF protocol	Displays whether OSPF is enabled or disabled.
# AS ext. routes	Displays the estimated number of Autonomous System external routes. The router cannot accept more than this number of AS external routes.
Estimated # routers	Displays the estimated number of routers found in the OSPF configuration.

Maximum LSA size	Displays the maximum size LSA that will be originated by this router.
External comparison	Displays the external route type used by OSPF when importing external information into the OSPF domain and when comparing OSPF external routes to RIP/BGP routes.
RFC 1583 compatibility	Indicates whether or not OSPF AS external route is compatible with RFC 1583.
AS boundary capability	Displays whether the router will import external routes into the OSPF domain.
Import external	Displays which routes will be imported.
Orig default route	Displays whether the router will import a default into the OSPF domain. When the value is “YES”, and a non-zero network number is displayed in parentheses. This indicates that the default route will be originated only if a route to that network is available.
Default route cost	Displays the cost and type that will be used in the imported default route.
Default forward addr	Displays the forwarding address that will be used for the originated default route.
Demand circuits	Displays whether demand circuit processing is supported.
External comparison	Displays the external route type used by OSPF when importing external information into the OSPF domain and when comparing OSPF external routes to RIP/BGP routes.
Inter-area multicast	Displays whether IP multicast datagrams will be forwarded between areas.
Area-ID	Displays the attached area ID (area summary information)
AuType	Displays the method used for area authentication. “Simple-pass” means a simple password scheme is being used for the area’s authentication.
Stub area	Displays whether or not the area being summarized is a stub area. Stub areas do not carry external routes, resulting in a smaller routing database. However, stub areas cannot contain AS boundary routers, nor can they support configured virtual links.
OSPF interfaces	For each interface, its IP address is printed, together with configured parameters. “Area” is the OSPF area to which the interface attaches. “Cost” indicates the TOS 0 cost (or metric) associated with the interface. “Rtrns” is the retransmission interval, which is the number of seconds between retransmissions of unacknowledged routing information. “TrnsDly” is the transmission delay, which is an estimate of the number of seconds it takes to transmit routing information over the interface (it must be greater than 0). “Pri” is the interface’s Router Priority, which is used when selecting the designated router. “Hello” is the number of seconds between Hello Packets sent out the interface. “Dead” is the number of seconds after Hellos cease to be heard that the router is declared down.
Virtual links	Lists all virtual links that have been configured with this router as end-point. “Virtual endpoint” indicates the OSPF Router ID of the other end-point. “Transit area” indicates the non-backbone area through which the virtual link is configured. Virtual links are considered treated by the OSPF protocol similarly to point-to-point networks. The other parameters listed in the command (“Rtrns”, “TrnsDly”, “Hello,” and “Dead”) are maintained for all interfaces. See the OSPF list interfaces command for more information.

areas Lists all information concerning configured OSPF areas.

Example: `list areas`

```
                                --Area configuration--
Area ID          AuType          Stub? Default-cost Import-summaries?
0.0.0.0          0=None           No         N/A           N/A
0.0.0.1          1=Simp-Pass      No         N/A           N/A
```

Area-ID	Displays the attached area ID (area summary information).
AuType	Displays the method used for area authentication. “Simple-pass” means a simple password scheme is being used for the area’s authentication.

Stub area	Displays whether or not the area being summarized is a stub area. Stub areas do not carry external routes, resulting in a smaller routing database. However, stub areas cannot contain AS boundary routers, nor can they support configured virtual links.
Default-cost	For stub areas the cost of the default to be originated as an OSPF summary (type 3) Link State Advertisement (LSA). For transit areas (for example, non-stub areas), this field is N/A.
Import-summaries	For stub areas, indicates whether or not OSPF summary (type 3) Link State Advertisements are to be originated into the stub area. This question does not apply to the default summary. For transit areas (for example, non-stub areas, this field is N/A.

interfaces

For each interface, its IP address is printed, together with configured parameters. “Area” is the OSPF area to which the interface attaches. “Cost” indicates the TOS 0 cost (or metric) associated with the interface. “Rtrns” is the retransmission interval, which is the number of seconds between retransmissions of unacknowledged routing information. “TrnsDly” is the transmission delay, which is an estimate of the number of seconds it takes to transmit routing information over the interface (it must be greater than 0). “Pri” is the interface’s router priority, which is used when selecting the designated router. “Hello” is the number of seconds between Hello Packets sent out the interface. “Dead” is the number of seconds after Hellos cease to be heard that the router is declared down.

Example: `list interfaces`

```
OSPF Config>list interface

                        --Interface configuration--
IP address        Area            Auth   Cost  Rtrns  Delay  Pri  Hello   Dead
200.1.1.2         0.0.0.2           0      10      5     1     1     10     40
10.69.1.2         0.0.0.0           1       1      5     1     1     10     40
OSPF Config>list virtual-link

                        --Virtual link configuration--
Virtual endpoint     Transit area    Auth   Rtrns  Delay Hello  Dead
4.4.4.4              0.0.0.1           1       10    5     30   180
10.1.1.2             0.0.0.1           1       10    5     30   180
OSPF Config>
OSPF Config>list area

                --Area configuration--
Area ID         Stub? Default-cost Import-summaries?
0.0.0.2          No         N/A           N/A
0.0.0.0          No         N/A           N/A
0.0.0.1          No         N/A           N/A
0.0.0.3          Yes        10            Yes
```

Note: Multicast parameters are not displayed if multicast is disabled. Demand circuit parameters are not displayed if none of the interfaces are configured as demand circuits.

neighbors

Lists neighbors to non-broadcast networks. It displays IP address of the neighbor and the IP address of the interface to that neighbor. It also indicates whether the neighbor is eligible to become the “Designated Router” on the net.

Example: `list neighbors`

```
                --Neighbor configuration--
Neighbor Addr     Interface Address   DR eligible?      Alternate TOS 0 Cost
2.3.4.5           1.2.3.4             yes               0
2.5.6.7           5.6.7.8             no                100
```

non-broadcast

Lists all information related to interfaces connected to non-broadcast multi-access networks. For each non-broadcast interface, as long as the

router is eligible to become designated router on the attached network, the polling interval is displayed together with a list of the router's neighbors on the non-broadcast network.

Example: `list non-broadcast`

```
        --NBMA configuration--
Interface Addr      Poll Interval
128.185.235.34      120
```

virtual-links

Lists all virtual links that have been configured with this router as end-point. "Virtual endpoint" indicates the OSPF router ID of the other end-point. "Transit area" indicates the non-backbone area through which the virtual link is configured. Virtual links are considered treated by the OSPF protocol similarly to point-to-point networks. The other parameters listed in the command ("Rtrns", "TrnsDly", "Hello," and "Dead") are maintained for all interfaces. See the OSPF **list interfaces** command for more information.

Example: `list virtual-links`

```
          --Virtual link configuration--
Virtual endpoint   Transit area   Rtrns  TrnsDly Hello Dead
0.0.0.0            0.0.0.1         10      5     30    180
```

Set

Use the **set** command to display or change the configuration information concerning OSPF areas, interfaces, non-broadcast networks, or virtual links. This command also allows you to set the way in which OSPF routes are compared to information obtained from other routing protocols.

Syntax:

set area
comparison
interface
non-broadcast
virtual-link

area Sets the parameters for an OSPF area. If no areas are defined, the router software assumes that all the router's directly attached networks belong to the backbone area (area ID 0.0.0.0).

Example: `set area`

```
Area number [0.0.0.0]? 0.0.0.1
Is this a stub area? [No]:  yes
Stub default cost? [0]:
Import summaries? [Yes]:
```

- *Area number* - is the OSPF area address.
- *Stub area designation.* If you designate "Yes":
  - The area does not receive any AS external link advertisements, reducing the size of your database and decreasing memory usage for routers in the stub area.
  - You cannot configure virtual links through a stub area.
  - You cannot configure a router within the stub area as an AS boundary router.

  External Routing in Stub Areas. You cannot configure the backbone as a stub area. External routing in stub areas is based on a default route. Each border area router that is attached to a stub area originates a

default route for this purpose. The cost of this default route is also configurable with the **set area** command.

comparison

Tells the router where the BGP/RIP/static routes fit in the OSPF hierarchy. The two lower levels consist of the OSPF internal routes. OSPF internal routes take precedence over information gained from any other sources, all of which are located on a single level.

Example: `set comparison`

```
OSPF Config> set comparison
Compare to type 1 or 2 externals [2]?
```

interface

Sets the OSPF parameters for the router's network interfaces.

1. The *interface IP address* is for each interface in the router.
2. *attaches to area* is the area to which the interface attaches.
3. The timer values are the same values for all routers attached to a common network segment.
   a. The *retransmission interval* is the interval after which a Link Request for one or more link state advertisements will be resent.

      Valid values: 1 to 65535 seconds

      Default Value: 5

   b. The *Transmission delay* is an estimate of the number of seconds that it takes to transmit link-state information over the interface.

      Each link-state advertisement has a finite lifetime that is equal to the constant MaxAge (1 hour). As each link-state advertisement is sent to the particular interfaces, it is aged by this configured transmission delay. The minimum delay is 1 second.

      Valid Values: 1 to 65535 seconds

      Default Value: 1

   c. The *Hello Interval* is the interval between Hello packets sent on the interface.

      Valid Values: 1 to 65535 seconds

      Default Value: 10

   d. The *Dead Router Interval*

      Dead Router Interval is the interval after which a router that has not sent a Hello will be considered dead. The Dead Router Interval defaults to four times the configured Hello Interval. The value for this parameter must be greater than the Hello Interval.

      Valid Values: 2 to ≥ 65535 seconds

      Default Value: 40 (or four times the configured Hello interval)
4. The *Router Priority* value is used for broadcast and non-broadcast multi–access networks to elect the designated router. For point-to-point links, this value should be **0**, which means that this router must not be elected the designated router for its network.

   Valid Values: 0 to 255

   Default Value: 1
5. The *Type of service 0 cost* is cost that will be used for the interface when the shortest path routes are computed for the area..

   Valid Values: 1 to 65534

   Default Value: 1

6. The *Demand Circuit* indicates whether or not the interface will be treated as a demand circuit for purposes of flooding LSAs (Link State Advertisements). Over demand circuits, LSAs will be flooded with the DoNotAge bit set over this interface and will not be flooded unless there is an actual change to the LSA. Refer to RFC 1793 for more information.

   Valid Values: Yes or No

   Default Value: No

7. The *Hello Suppression* indicates whether or not Hello packets will be suppressed on the interface once the neighbors reach the full state. Demand circuits must be enabled on the interface for Hello Suppression to be requested or allowed. Currently, Hello Suppression is only supported on ATM and ISDN Dial-on-Demand links. Refer to RFC 1793 for more information.

   Valid Values: Allow, Request, or Disable

   Default Value: Allow

Allow	Allows a neighbor to request Hello Suppression.
Request	Requests Hello Suppression from a neighbor.
Disable	Disables Hello Suppression and continues sending Hellos.

8. The *Demand Circuit Down Poll Interval* indicates the duration between hello polls sent when there is a failure to send data on a demand circuit with hello suppression active. Currently, hello suppression is only supported on ATM and ISDN Dial-on-Demand links. Refer to RCF 1793 for more information.

   Valid Values: 1 to 65535

   Default Value: 60

9. The *Authentication type* defines the authentication procedure to be used for OSPF packets on the interface. The choices are 1, which indicates a simple password; or 0, which indicates that no authentication is necessary to exchange OSPF packets on the interface. When 1 is specified, the authentication key must also be specified.

   Valid Values: 0, 1

   Default Value: 0

10. The *Authentication key* is the parameter that defines the password used for this OSPF area. When password authentication is used, only packets with the correct authentication key are accepted.

    Valid Values: any 1-8 characters

    Default Value: a null string

Example: `set interface`

```
Interface IP address [0.0.0.0]? 10.69.1.2
Attaches to area [0.0.0.0]?
Retransmission Interval (in seconds) [5]?
Transmission Delay (in seconds) [1]? 1
Router Priority [1]? 1
Hello Interval (in seconds) [10]?
Dead Router Interval (in seconds) [40]?
Type Of Service 0 cost [1]?
Demand Circuit (Yes or NO) ?[No]:
Authentication Type (0 - none, 1 - simple) [0]? 1
Authentication Key []? AceeOSPF
Retype Auth.  Key []? AceeOSPF
```

When responding to the prompts, supply the IP address for each interface in the router and answer the questions that follow. For the following parameters, you must enter the same value for all routers attached to a common network:

- Hello interval
- Dead router interval
- Authentication key (if an authentication of 1 is used)

The first prompt asks for the OSPF area to which the interface attaches. For example, suppose that the interface address mask is 255.255.255.0, indicating that the interface attaches to a subnet (128.185.138.0) of network 128.185.0.0. All other OSPF routers attached to subnet 128.185.138.0 must also have their *Hello interval* set to 10, *dead router interval* set to 40, and their interface *authentication key* set to xyz_q.

Note that IP interfaces to point-to-point lines may be unnumbered. In this case a net index is configured instead of an IP address. This implementation of OSPF will work with these unnumbered interfaces, but to work correctly, both ends of the point-to-point line must use an unnumbered interface.

non-broadcast

Overrides the point-to-multipoint default to select NBMA for X.25, Frame Relay or ATM networks. This parameter specifies the interval that determines the frequency of Hellos sent to neighbors that are inactive. You must set non-broadcast consistently across all interfaces that attach to the same subnetwork for OSPF to function correctly.

For Frame Relay or ATM networks, however, the **set non-broadcast** command is used to configure an OSPF interface as connecting to a non-broadcast multi-access network. If the **set non-broadcast** command is not used, the interface is assumed to be connected to a point-to-multipoint network. In Frame Relay networks, all OSPF interfaces must be configured as connecting to the same type of network (non-broadcast multi-access or point-to-multipoint), so if the **set non-broadcast** command is used for one router's interface, it must be configured on the interfaces for all routers attaching to the network.

Example: set non-broadcast

```
Interface IP address [0.0.0.0]? 128.185.138.19
Poll Interval [120]
```

The *interface IP address* has:

Valid Values: Any valid IP address.

Default Value: none

The NBMA Poll Interval is used to send Hello packets to inactive neighbors. (Inactive neighbors are those neighbors that the router has not heard from for a period greater than the Dead Router interval.) The router still polls these neighbors at a reduced rate. Set the NBMA Poll Interval much higher than the configured Hello Interval for the router.

Valid Values: 1 to 65535 seconds

Default Value: 120 seconds

Example: `set non-broadcast`

```
Interface IP address [0.0.0.0]? 128.185.138.19
Poll Interval [120]?
```

virtual-link

Configures virtual links between any two area border routers. To maintain

backbone connectivity you must have all of your backbone routers interconnected either by permanent or virtual links. Virtual links are considered to be separate router interfaces connecting to the backbone area. Therefore, you are asked to also specify many of the interface parameters when configuring a virtual link.

Virtual links can be configured between any two backbone routers that have an interface to a common non-backbone area. Virtual links are used to maintain backbone connectivity and must be configured at both end-points.

Note: This OSPF implementation supports the use of virtual links when one end of the virtual link may be an unnumbered point to point line. For this configuration to work, the router id must be used as the source address in OSPF protocol messages sent over the virtual link. Use of the router id can be insured by configuring the internal IP address with the address used as the router id. Another requirement for this configuration to work is that the OSPF implementations at both ends of the virtual link support it.

1. The *virtual endpoint (router ID)* defines the ID of the virtual neighbor.

   Valid Values: Any valid IP address.

   Default Value: none

2. The *link's transit area*. is the non-backbone, non-stub area through which the virtual link is configured. Virtual links can be configured between any two area border routers that have an interface to a common non-backbone and non-stub area. Virtual links must be configured in each of the link's two end-points.

   Valid Values: 0.0.0.1 to 255.255.255.255

   Default Value: 0.0.0.1

3. The timer values are the same values for all routers attached to a common network segment.

   a. The *retransmission interval* is the interval after which a Link Request for one or more link state advertisements will be resent.

      Valid Values: 1 to 65535 seconds

      Default Value: 10

   b. The *Transmission delay* parameter is an estimate of the number of seconds that it takes to transmit link-state information over the interface.

      Each link-state advertisement has a finite lifetime that is equal to the constant MaxAge (1 hour). As each link-state advertisement is sent to the particular interfaces, it is aged by this configured transmission delay. The minimum delay is 1 second.

      Valid Values: 1 to 65535 seconds

      Default Value: 5

   c. The *Hello Interval* is the interval between Hello packets sent on the interface.

      Valid Values: 1 to 255 seconds

      Default Value: 30

   d. The *Dead Router Interval* is the interval after which a router that has not sent a Hello will be considered dead. This

parameter defaults to six times the configured Hello Interval and must be set to a value greater than the Hello Interval.

Valid Values: 2 to 65535 seconds

Default Value: 180

4. The *Authentication type* defines the authentication procedure to be used for OSPF packets on the virtual link. The choices are 1, which indicates a simple password; or 0, which indicates that no authentication is necessary to exchange OSPF packets on the interface. When 1 is specified, the authentication key must also be specified.

   Valid Values: 0, 1

   Default Value: 0

5. The *Authentication key*. defines the password used for this OSPF area. When password authentication is used, only packets with the correct authentication key are accepted.

Valid Values: any 1-8 characters

Default Value: a null string

Example: `set virtual-link`

```
Virtual endpoint  (Router ID) [0.0.0.0]? 10.1.1.2
Link's transit area [0.0.0.1]?
Virtual link already exists - record will be modified.
Retransmission Interval (in seconds) [10]?
Transmission Delay (in seconds) [5]?
Hello Interval (in seconds) [30]?
Dead Router Interval (in seconds) [180]?
Authentication Type (0 - none, 1 - simple) [0] 1
Authentication Key []? AceeOSPF
Retype Auth. Key []? AceeOSPF
```

Accessing the OSPF Monitoring Environment

Use the following procedure to access the OSPF monitoring commands. This process gives you access to the OSPF *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. For example:

   ```
   * talk 5
   +
   ```

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.

2. At the + prompt, enter the **protocol ospf** command to get you to the `OSPF>` prompt.

   Example:

   ```
   + prot ospf
   OSPF>
   ```

OSPF Monitoring Commands

This section summarizes and then explains all the OSPF monitoring commands. These commands enable you to monitor the OSPF routing protocol. Table 68 on page 479 lists the OSPF monitoring commands.

Enter the OSPF monitoring commands at the OSPF> prompt.

Table 68. OSPF Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Advertisement	Displays a link state advertisement belonging to the OSPF database.
Area summary	Displays OSPF area statistics and parameters.
AS external	Lists the AS external advertisements belonging to the OSPF link state database.
Database summary	Displays the advertisements belonging to an OSPF area’s link state database.
Dump routing tables	Displays the OSPF routes contained in the routing table.
Interface summary	Displays OSPF interface statistics and parameters.
Neighbor summary	Displays OSPF neighbor statistics and parameters.
Ping	Continuously sends ICMP Echo Requests (or pings) a given destination, printing a line for each response received.
Policy	Displays any configured AS boundary router import policy.
Reset	Resets the OSPF configuration dynamically.
Routers	Displays the reachable OSPF area-border routers and AS-boundary routers.
Size	Displays the number of LSAs currently in the link state database, categorized by type.
Statistics	Displays OSPF statistics detailing memory and network usage.
Traceroute	Displays the complete route (hop-by-hop) to a given destination.
Weight	Dynamically changes the cost of an OSPF interface.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Advertisement Expansion

Use the **advertisement expansion** command to print the contents of a link state advertisement contained in the OSPF database. For a summary of the router’s advertisements use the **database** command.

A link state advertisement is defined by its link state type, link state ID and its advertising router. There is a separate link state database for each OSPF area. Providing an area-id on the command line tells the software which database you want to search. The different kinds of advertisements, which depend on the value given for link-state-type, are:

- Router links - Contain descriptions of a single router’s interface.
- Network links - Contain the list of routers attached to a particular interface.
- Summary nets - Contain descriptions of a single inter-area route.
- Summary AS boundary routers - Contain descriptions of the route to an AS boundary router in another area.
- AS external nets - Contain descriptions of a single route.
- Multicast group memberships - Contain descriptions of a particular group’s membership in the neighborhood of the advertising router.

Note: Link State IDs, advertising routers (specified by their router IDs), and area IDs take the same format as IP addresses. For example, the backbone area can be entered as 0.0.0.0.

Example 1 shows an expansion of a router links advertisement. The router's ID is 128.185.184.11. It is an AS boundary router and has three interfaces to the backbone area (all of cost 1). Detailed field descriptions are provided with the example.

This command has also been enhanced in two ways. First of all, when displaying router-LSAs and network-LSAs, the reverse cost of each router-to-router link and router-to-transit-network link is displayed, as well as the previously displayed forward cost. This is done because routing of multicast datagrams whose source lies in different areas/Autonomous systems is based on reverse cost instead of forward cost. In those cases where there is no reverse link (which means that the link will never be used by the Dijkstra), the reverse cost is shown as "1-way".

In addition, the LSA's OSPF options are displayed in the same manner as they were displayed in the detailed OSPF **neighbor** command.

New group-membership-LSAs can also be displayed. The "LS destination" of each group-membership-LSA is a group address. A router originates a group-membership-LSA for each group that has members on one or more of the router's attached networks. The group-membership-LSA for the group lists those attached transit networks having group members (the type "2" vertices), and when there are members belonging to one or more attached stub networks, or if the router itself is a member of the multicast group, a type "1" vertex whose ID is the router's OSPF router ID is included.

Syntax:

advertisement *ls-type link-state-id advertising-router area-id*

Example 1: `advertisement 1 128.185.184.11 0.0.0.0`

```
LS age:     173
LS options:  E,MC,DC
LS type:     1
LS destination (ID): 128.185.184.11
LS originator:   128.185.184.11
LS sequence no:  0x80000047
LS checksum:     0x122
LS length:       60
Router type:  ASBR,W
# router ifcs:   3
         Link ID:          128.185.177.31
         Link Data:        128.185.177.11
         Interface type:   2
                  No. of metrics: 0
                  TOS 0 metric:   3 (0)
         Link ID:          128.185.142.40
         Link Data:        128.185.142.11
         Interface type:   2
                  No. of metrics: 0
                  TOS 0 metric:   4 (0)
         Link ID:          128.185.184.0
         Link Data:        255.255.255.0
         Interface type:   3
                  No. of metrics: 0
                  TOS 0 metric:   1
```

LS age Indicates the age of the advertisement in seconds.

LS options	Indicates the optional OSPF capabilities supported by the OSPF object corresponding to the advertisement. These capabilities include: **E** Indicates that type 5 (external advertisements) are supported in area corresponding to the advertisement. This is always set for type 5 (external advertisements). **T** Routing based on IP TOS (Type of Service) is supported. **DC** Demand circuits are supported as described in RFC 1793.
LS type	Classifies the advertisement and dictates its contents: 1 (router links advertisement), 2 (network link advertisement), 3 (summary link advertisement), 4 (summary ASBR advertisement), 5 (AS external link) and 6 (group-membership advertisement).
LS destination	Identifies what is being described by the advertisement. Depends on the advertisement type. For router links and ASBR summaries, it is the OSPF router ID. For network links, it is the IP address of the network's designated router. For summary links and AS external links, it is a network/subnet number. For group-membership advertisements, it is a particular multicast group.
LS originator	OSPF router ID of the originating router.
LS sequence number	Used to distinguish separate instances of the same advertisement. Should be looked at as a signed 32-bit integer. Starts at 0x80000001, and increments by one each time the advertisement is updated.
LS checksum	A checksum of advertisement contents, used to detect data corruption.
LS length	The size of the advertisement in bytes.
Router type	Indicates the level of function of the router. ASBR means that the router is an AS boundary router, ABR that the router is an area border router, and W that the router is a wildcard multicast receiver.
# Router ifcs	The number of router interfaces described in the advertisement.
Link ID	Indicates what the interface connects to. Depends on Interface type. For interfaces to routers (i.e., point-to-point links), the Link ID is the neighbor's router ID. For interfaces to transit networks, it is the IP address of the network designated router. For interfaces to stub networks, it is the network's network/subnet number.
Link Data	4 bytes of extra information concerning the link, it is either the IP address of the interface (for interfaces to point-to-point networks and transit networks), or the subnet mask (for interfaces to stub networks).
Interface type	One of the following: 1 (point-to-point connection to another router, 2 (connection to transit network), 3 (connection to stub network) or 4 (virtual link).
No. of metrics	The number of non-zero TOS values for which metrics are provided for this interface.
TOS 0 metric	The cost of the interface. In parenthesis the reverse cost of the link is given (derived from another advertisement). If there is no reverse link, "1-way" is displayed.

The LS age, LS options, LS type, LS destination, LS originator, LS sequence no, LS checksum and LS length fields are common to all advertisements. The Router type and # router ifcs are seen only in router links advertisements. Each link in the router advertisement is described by the Link ID, Link Data, and Interface type fields. Each link can also be assigned a separate cost for each IP Type of Service (TOS); this is described by the No. of metrics and TOS 0 metric fields (the router currently does not route based on TOS, and looks at the TOS 0 cost only).

Example 2 shows an expansion of a group-membership advertisement. A group-membership advertisement for a given group/advertising router combination lists those networks directly attached to the advertising router which have group

members. It also lists whether the router itself is a member of the specified group. The example below shows that network 128.185.184.0 has members of group 224.0.1.1.

Example 2: `adv 6 224.0.1.1 128.185.184.114`

```
For which area [0.0.0.0]?

LS age:     168
LS options:  E
LS type:     6
LS destination (ID): 224.0.1.1
LS originator:   128.185.184.114
LS sequence no:  0x80000001
LS checksum:     0x7A3
LS length:       28
Vertex type: 2
Vertex ID:   128.185.184.114
```

Vertex type — Describes the object having group members, one of: 1 (the router itself, or stub networks attached to the router) or 2 (a transit network).

Vertex ID — When the vertex type is 1, always the advertising router''s ID. When the vertex type is 2, the IP address of the transit network's designated router.

Area Summary

Use the **area summary** command to display the statistics and parameters for all OSPF areas attached to the router.

In the example below, the router attaches to a single area (the backbone area). A simple password scheme is being used for the area's authentication. The router has three interfaces attaching to the area, and has found 4 transit networks, 7 routers and no area border routers when doing the SPF tree calculation for the backbone.

Syntax:

area

Example:

```
Area ID           #ifcs  #nets  #rtrs  #brdrs
0.0.0.1               1      1      2       2
0.0.0.0               3      0      3       2
```

ifcs — Indicates the number of router interfaces attached to the particular area. These interfaces are not necessarily functional.

nets — Indicates the number of transit networks that have been found while doing the SPF tree calculation for this area.

rtrs — Indicates the number of routers that have been found when doing the SPF tree calculation for this area.

brdrs — Indicates the number of area border routers that have been found when doing the SPF tree calculation for this area.

AS-external advertisements

Use the **AS-external advertisements** command to list the AS external advertisements belonging to the OSPF routing domain. One line is printed for each advertisement. Each advertisement is defined by the following three parameters: its link state type (always 5 for AS external advertisements), its link state ID (called the LS destination), and the advertising router (called the LS originator).

Syntax:

as-external

Example: `as-external`

```
Type LS-destination  LS-originator   Seq-Number  Age Unreach Xsum   Options
  5  10.13.64.0      10.1.62.1       0x80000385 1422         0x7791 E,DC
  5  10.14.64.0      10.1.62.1       0x80000385 1420         0x6B9C E,DC

               # advertisements:       2
               Checksum total:         0xE32D
```

Type	Always 5 for AS external advertisements.
LS destination	Indicates an IP network/subnet number. These network numbers belong to other Autonomous Systems.
LS originator	Advertising router.
Unreach	Indicates how long the destination associated with a Link State Advertisement (LSA) that is DoNotAge has been unreachable. If the LSA is DoNotAge, *DA* will appear after the Age column before the Unreach column. If the LSA is **not** DoNotAge, there will be blanks.
Seqno, Age, Xsum	It is possible for several instances of an advertisement to be present in the OSPF routing domain at any one time. However, only the most recent instance is kept in the OSPF link state database (and printed by this command). The LS sequence number (Seqno), LS age (Age) and LS checksum fields (Xsum) are compared to see which instance is most recent. The LS age field is expressed in seconds. Its maximum value is 3600.
Options	These are the Link State Options, which are the optional OSPF capabilities supported by the OSPF object corresponding to the advertisement. These capabilities include: **E** Indicates that type 5 (external advertisements) are supported in area corresponding to the advertisement. This is always set for type 5 (external advertisements). **T** Routing based on IP TOS (Type of Service) is supported. **DC** Demand circuits are supported as described in RFC 1793.

At the end of the display, the total number of AS external advertisements is printed, along with a checksum total over all of their contents. The checksum total is simply the 32-bit sum (carries discarded) of the individual advertisement's LS checksum fields. This information can be used to quickly determine whether two OSPF routers have synchronized databases.

Database Summary

Use the **database summary** command to display a description of the contents of a particular OSPF area's link state database. AS external advertisements are omitted from the display. A single line is printed for each advertisement. Each advertisement is defined by the following three parameters: its link state type (called Type), its link state ID (called the LS destination) and the advertising router (called the LS originator).

Syntax:

database *area-id*

Example: `database 0.0.0.0`

```
Type LS-destination  LS-originator   Seq-Number  Age Unreach Xsum   Options
  1  10.1.62.1       10.1.62.1       0x80004963  496         0xBC15 E,DC
  1  10.1.62.2       10.1.62.2       0x800250FF    6         0xCA6F E,DC
  ⋮
```

```
# advertisements:       99
Checksum total:         0x2CD102
```

Type	Separate LS types are numerically displayed: type 1 (router links advertisements), type 2 (network links advertisements), type 3 (network summaries), type 4 (AS boundary router summaries), and type 6 (group-membership-LSAs).
LS destination	Indicates what is being described by the advertisement.
LS originator	Advertising router.
Unreach	Indicates how long the destination associated with a Link State Advertisement (LSA) that is DoNotAge has been unreachable. If the LSA is DoNotAge, *DA* will appear after the Age column before the Unreach column. If the LSA is **not** DoNotAge, there will be blanks.
Seqno, Age, Xsum	It is possible for several instances of an advertisement to be presenting the OSPF routing domain at any one time. However, only the most recent instance is kept in the OSPF link state database (and printed by this command). The LS sequence number (Seqno), LS age (Age) and LS checksum fields (Xsum) are compared to see which instance is most recent. The LS age field is expressed in seconds. Its maximum value is 3600.
Options	These are the Link State Options, which are the optional OSPF capabilities supported by the OSPF object corresponding to the advertisement. These capabilities include:
	E Indicates that type 5 (external advertisements) are supported in area corresponding to the advertisement. This is always set for type 5 (external advertisements).
	T Routing based on IP TOS (Type of Service) is supported.
	DC Demand circuits are supported as described in RFC 1793.

At the end of the display, the total number of advertisements in the area database is printed, along with a checksum total over all of their contents. The checksum total is simply the 32-bit sum (carries discarded) of the individual advertisement's LS checksum fields. This information can be used to quickly determine whether two OSPF routers have synchronized databases.

Dump Routing Tables

Use the **dump routing tables** command to display all the routes that have been calculated by OSPF and are now present in the routing table. Its output is similar in format to the IP monitoring's dump routing tables command.

Syntax:

dump

Example: `dump`

```
Type   Dest net         Mask     Cost Age   Next hop(s)
SPE1   0.0.0.0          00000000  4    3    128.185.138.39
SPF*   128.185.138.0    FFFFFF00  1    1    Eth/0
Sbnt   128.185.0.0      FFFF0000  1    0    None
SPF    128.185.123.0    FFFFFF00  3    3    128.185.138.39
SPF    128.185.124.0    FFFFFF00  3    3    128.185.138.39
SPF    192.26.100.0     FFFFFF00  3    3    128.185.131.10
RIP    197.3.2.0        FFFFFF00  10   30   128.185.131.10
RIP    192.9.3.0        FFFFFF00  4    30   128.185.138.21
Del    128.185.195.0    FFFFFF00  16   270  None

Default gateway in use.
```

```
  Type Cost Age  Next hop
  SPE1 4    3    128.185.138.39

  Routing table size: 768 nets (36864 bytes), 36 nets known
```

Type (route type)	Indicates how the route was derived. Sbnt - Indicates that the network is subnetted; such an entry is a place-holder only. Dir - Indicates a directly connected network or subnet. RIP - Indicates the route was learned through the RIP protocol. Del - Indicates the route has been deleted. Stat - Indicates a statically configured route. BGP - Indicates routes learned through the BGP protocol. BGPR - Indicates routes learned through the BGP protocol that are readvertised by OSPF and RIP. Fltr - Indicates a routing filter. SPF - Indicates that the route is an OSPF intra-area route. SPIA - Indicates that it is an OSPF inter-area routes. SPE1, SPE2 - Indicates OSPF external routes (type 1 and 2 respectively). Rnge - Indicates a route type that is an active OSPF area address range and is not used in forwarding packets.
Dest net	IP destination network/subnet.
Mask	IP address mask.
Cost	Route Cost.
Age	For RIP and BGP routes, the time that has elapsed since the routing table entry was last refreshed.
Next Hop	IP address of the next router on the path toward the destination host. Also displayed is the interface type used by the sending router to forward the packet.

An asterisk (*) after the route type indicates the route has a static or directly connected backup. A percent sign (%) after the route type indicates that RIP updates will always be accepted for this network/subnet.

A number in parentheses at the end of the column indicates the number of equal-cost routes to the destination. The first hops belonging to these routes can be displayed with the IP monitoring's **route** command.

Interface Summary

Use the **interface summary** command to display statistics and parameters related to OSPF interfaces. If no arguments are given (see Example 1), a single line is printed summarizing each interface. If an interface's IP address is given (see Example 2), detailed statistics for that interface will be displayed.

Syntax:

interface *interface-ip-address*

Example 2: `interface 128.185.125.22`

OSPF Monitoring Commands (Talk 5)

```
Interface address:      128.185.125.22
Attached area:          0.0.0.1
Physical interface:     Eth/1
Interface mask:         255.255.255.0
Interface type:         Brdcst
State:                  32
Authentication Type:    None
Designated Router:      128.185.184.34
            Backup DR:              128.185.184.11

DR Priority:       1  Hello interval:   10  Rxmt interval:     5
Dead interval:    40  TX delay:          1  Poll interval:     0
Demand Circuit   off  Max pkt size:   2044  TOS 0 cost:        1

# Neighbors:       0  # Adjacencies:     0  # Full adjs.:      0
```

Interface Address	Interface IP address.
Attached Area	Attached area ID.
Physical interface	Displays physical interface type and number.
Interface Mask	Displays interface subnet mask.
Interface type	Can be either Brdcst (broadcast, e.g., an Ethernet interface), PP (a point-to-point network, e.g., a synchronous serial line), P-2-MP (point-to-multipoint, e.g., a Frame-Relay network), Multi (non-broadcast, multi-access, e.g., an X.25 connection) and VLink (an OSPF virtual link).
State	Can be one of the following: 1 (Down), 2 (Looped back), 4 (Waiting), 8 (Point-to-Point), 16 (DR other), 32 (Backup DR), 64 (Designated router) or 128 (Full).
Authentication Type	Indicates the type of authentication active for the interface. Supported types are none or simple.
Designated Router	IP address of the designated router.
Backup DR	IP address of the backup designated router.
DR Priority	Displays priority assigned to designated router.
Hello interval	Displays the current hello interval value.
Rxmt interval	Displays the current retransmission interval value.
Dead interval	Displays the current dead interval value.
TX delay	Displays the current transmission delay value.
Poll interval	Displays the current poll interval value.
Max pkt size	Displays the maximum size for an OSPF packet sent out this interface.
Demand circuit	Indicates whether or not demand circuit processing is active on the interface.
TOS 0 cost	Displays the interface's TOS 0 cost.
# Neighbors	Number of neighbors. This is the number of routers whose hellos have been received, plus those that have been configured.
# Adjacencies	Number of adjacencies. This is the number of neighbors in state Exchange or greater.
# Full adj	Number of full adjacencies. The number of full adjacencies is the number of neighbors whose state is Full (and therefore, with which the router has synchronized databases).
DL unicast	Displays whether multicast datagrams are to be forwarded as data-link multicasts or as data-link unicasts.
# MC data acc	Displays the number of multicast datagrams that have been successfully forwarded.
# MC data out	Displays the number of datagrams that have been forwarded out the interface (either as data-link multicasts or data-link unicasts).
Network Capabilities	Displays the network capabilities for the interface.
Nbr node: type and ID	Displays the identity of the upstream node if the router were supposed to receive datagrams on this interface. Type here is an integer from 1 to 3, with 1 indicating router, 2 indicating transit net and 3 indicating stub net.

Neighbor

Use the **neighbor** command to display statistics and parameters related to OSPF neighbors. If no arguments are given (see Example 1), a single line is printed summarizing each neighbor. If a neighbor's IP address is given (see Example 2), detailed statistics for that neighbor will be displayed.

Syntax:

neighbor

Example 1: `neighbor`

```
Neighbor addr   Neighbor ID     State  LSrxl  DBsum  LSreq  Ifc
128.185.125.39  128.185.136.39  128      0      0       0   PPP/1
128.185.125.41  128.185.128.41    8      0      0       0   PPP/1
128.185.125.38  128.185.125.38    8      0      0       0   PPP/1
128.185.125.25  128.185.129.25    8      0      0       0   PPP/1
128.185.125.40  128.185.129.40  128      0      0       0   PPP/1
128.185.125.24  128.185.126.24    8      0      0       0   PPP/1
```

Neighbor addr	Displays the neighbor address.
Neighbor ID	Displays the neighbor's OSPF router ID.
Neighbor State	Can be one of the following: 1 (Down), 2 (Attempt), 4 (Init), 8 (2-Way), 16 (ExStart), 32 (Exchange), 64 (Loading) or 128 (Full).
LSrxl	Displays the size of the current link state retransmission list for this neighbor.
DBsum	Displays the size of the database summary list waiting to be sent to the neighbor.
LSreq	Displays the number of more recent advertisements that are being requested from the neighbor.
Ifc	Displays the interface shared by the router and the neighbor.

Example 2: `neighbor 128.185.138.39`

The meaning of most of the displayed fields is given in section 10 of the OSPF specification (RFC 1583) .

```
Neighbor IP address:    128.185.184.34
OSPF Router ID:         128.185.207.34
Neighbor State:         128
Physical interface:     Eth/1
DR choice:              128.185.184.34
Backup choice:          128.185.184.11
DR Priority:            1
Nbr options:            E,MC
Alternate TOS 0 cost:   5

DB summ qlen:      0  LS rxmt qlen:      0  LS req qlen: 0
Last hello:        7  No Hello         Off

# LS rxmits:     108  # Direct acks:    13  # Dup LS rcvd: 572
# Old LS rcvd:     2  # Dup acks rcv:  111  # Nbr losses:   29
# Adj. resets:    30
```

Neighbor IP addr	Neighbor IP address.
OSPF router ID	Neighbor's OSPF router ID.
Neighbor State	Can be one of the following: 1 (Down), 2 (Attempt), 4 (Init), 8 (2-Way), 16 (ExStart), 32 (Exchange), 64 (Loading) or 128 (Full).
Physical interface	Displays physical interface type and number of the router and neighbor's common network.
DR choice, backup choice, DR priority	Indicate the values seen in the last hello received from the neighbor.

Nbr options	Indicates the optional OSPF capabilities supported by the neighbor. These capabilities are denoted by E (processes type 5 externals; when this is not set the area to which the common network belongs has been configured as a stub), T (can route based on TOS) and MC (can forward IP multicast datagrams). This field is valid only for those neighbors in state Exchng or greater.
DBsumm qlen	Indicates the number of advertisements waiting to be summarized in Database Description packets. It should be zero except when the neighbor is in state Exchange.
LS rxmt qlen	Indicates the number of advertisements that have been flooded to the neighbor, but not yet acknowledged.
LS req qlen	Indicates the number of advertisements that are being requested from the neighbor in state Loading.
Last hello	Indicates the number of seconds since a hello has been received from the neighbor.
# LS rxmits	Indicates the number of retransmissions that have occurred during flooding.
# direct acks	Indicates responses to duplicate link state advertisements.
# Dup LS rcvd	Indicates the number of duplicate retransmissions that have occurred during flooding.
# Old LS rcvd	Indicates the number of old advertisements received during flooding.
# Dup acks rcvd	Indicates the number of duplicate acknowledgments received.
# Nbr losses	Indicates the number of times the neighbor has changed to Down state.
# Adj. resets	Counts entries to state ExStart.

Ping

See “Ping” on page 362 for an explanation of the **Ping** command.

Reset

Use the OSPF **reset** command to dynamically modify the OSP routing configuration without restarting the router. For more information see “Dynamically Changing OSPF Configuration Parameters” on page 461.

Note: During a restart, OSPF routes will be retained in the routing table to maintain IP forwarding.

Syntax:

reset ospf

Example:

```
OSPF>interface

Ifc Address      Phys    assoc. Area     Type    State   Auth   #nbrs  #adjs
153.2.2.25       Eth/0   0.0.0.1         Brdcst   16     None       3      2
10.69.1.1        FR/0    0.0.0.0         P-2-MP    8     None       1      1

OSPF>
*t 6

OSPF Config>delete interface 10.69.1.1
OSPF Config>
*t 5

OSPF>reset ospf
OSPF>interface

Ifc Address      Phys    assoc. Area     Type    State   Auth   #nbrs  #adjs
153.2.2.25       Eth/0   0.0.0.1         Brdcst   16     None       3      2
```

Traceroute

See “Traceroute” on page 366 for an explanation of the **Traceroute** command.

Routers

Use the **routers** command to display all router routes that have been calculated by OSPF and are now present in the routing table. With the **dump routing tables** command, the Net field indicates that the destination is a network. The routers command covers all other destinations.

Syntax:

routers

Example:

```
DType RType Destination       AREA          Cost   Next hop(s)

 ASBR  SPF   128.185.142.9    0.0.0.1         1   128.185.142.9
 Fadd  SPF   128.185.142.98   0.0.0.1         1    0.0.0.0
 Fadd  SPF   128.185.142.7    0.0.0.1         1    0.0.0.0
 Fadd  SPF   128.185.142.48   0.0.0.1         1    0.0.0.0
 Fadd  SPF   128.185.142.111  0.0.0.1         1    0.0.0.0
 Fadd  SPF   128.185.142.38   0.0.0.1         1    0.0.0.0
 Fadd  SPF   128.185.142.11   0.0.0.1         1    0.0.0.0
   BR  SPF   128.185.142.9    0.0.0.2         1   128.185.142.9
   BR  SPF   128.185.142.9    0.0.0.2         2   128.185.184.114
 Fadd  SPF   128.185.142.47   0.0.0.2         1    0.0.0.0
```

DType — Indicates destination type:
- **Net** indicates that the destination is a network
- **ASBR** indicates that the destination is an AS boundary router
- **ABR** indicates that the destination is an area border router
- **Fadd** indicates a forwarding address (for external routes)

RType — Indicates route type and how the route was derived:
- **SPF** indicates that the route is an intra-area route (comes from the Dijkstra calculation)
- **SPIA** indicates that it is an inter-area route (comes from considering summary link advertisements).

Destination — Destination router’s OSPF ID. For Type D entries, one of the router’s IP addresses is displayed (which corresponds to a router in another AS).

Area — Displays the AS area to which it belongs.

Cost — Displays the route cost.

Next hop — Address of the next router on the path toward the destination host. A number in parentheses at the end of the column indicates the number of equal-cost routes to the destination.

Size

Use the **size** command to display the number of LSAs currently in the link state database, categorized by type.

Syntax:

size

Example:

```
# Router-LSAs:            6
# Network-LSAs:           2
# Summary-LSAs:           45
# Summary Router-LSAs:    6
# AS External-LSAs:       2
# Group-membership-LSAs:  11

# Intra-area routes:      11
# Inter-area routes:      15
# Type 1 external routes: 0
# Type 2 external routes: 2
```

Statistics

Use the **statistics** command to display statistics generated by the OSPF routing protocol. The statistics indicate how well the implementation is performing, including its memory and network utilization. Many of the fields displayed are confirmation of the OSPF configuration.

Syntax:

statistics

Example:

```
OSPF>statistics

               OSPF Router ID:         1.1.1.1
               External comparison:    Type 2
               RFC 1583 compatibility: Yes
                           Demand circuit support: Yes
               AS boundary capability: No
               Import external routes: None
               Orig. default route:    No (0,0.0.0.0)
               Default route cost:     (1, Type 2)
               Default forward. addr:  0.0.0.0

  Attached areas:                   1  Estimated # external routes:   1000
  Estimated # OSPF routers:        50  Estimated heap usage:        148000
  OSPF packets rcvd:               63  OSPF packets rcvd w/ errs:        1
  Transit nodes allocated:         21  Transit nodes freed:             17
  LS adv. allocated:               83  LS adv. freed:                   61
  Queue headers alloc:             64  Queue headers avail:             64
  Maximum LSA size:              2048

  # Dijkstra runs:                  7  Incremental summ. updates:        2
  Incremental VL updates:            0  Buffer alloc failures:           0
  Multicast pkts sent:              31  Unicast pkts sent:              19
  LS adv. aged out:                  9  LS adv. flushed:                11
  Ptrs To Invalid LS adv:            0  Incremental ext. updates:       14
  LSA Max Random Initial Age:     1770  LSA MINARRIVAL rejects:          1
  External LSA database:
  Current state:                Normal
  Number of LSAs:                   10  Number of overflows:             0
```

OSPF Router ID	Displays the router's OSPF ID.
External comparison	Displays the external route type used by the router when importing external routes.
RFC 1583 compatibility	Indicates whether or not OSPF AS external route computation will be compatible with RFC 1583.
AS boundary capability	Displays whether external routes will be imported.
Import external routes	Displays which external routes will be imported.
Orig default route	Displays whether the router will advertise an OSPF default route. If the value is "Yes" and a nonzero number is displayed in parentheses, then a default route will be advertised only when a route to the network exists.
Default route cost	Displays the cost and type of the default route (if advertised).

Default forward addr	Displays the forwarding address specified in the default route (if advertised).
Attached areas	Indicates the number of areas that the router has active interfaces to.
Estimated heap usage	Rough indication of the size of the OSPF link state database (in bytes).
Transit nodes	Allocated to store router links and network links advertisements.
LS adv.	Allocated to store summary link and AS external link advertisements.
Queue headers	Form lists of link state advertisements. These lists are used in the flooding and database exchange processes; if the number of queue headers allocated is not equal to the number freed, database synchronization with some neighbor is in progress.
# Dijkstra runs	Indicates how many times the OSPF routing table has been calculated from scratch.
Maximum LSA size	The maximum size LSA that can be originated by this router. This is the minimum of the value configured through OSPF configuration and the maximum packet size computed or configured through general configuration.
Incremental summ updates, incremental VL updates	Indicate that new summary link advertisements have caused the routing table to be partially rebuilt.
Buffer alloc failures.	Indicate buffer allocation failures. The OSPF system will recover from temporary lack of packet buffers.
Multicast pkts sent	Covers OSPF hello packets and packets sent during the flooding procedure.
Unicast pkts sent	Covers OSPF packet retransmissions and the Database Exchange procedure.
LS adv. aged out	Counts the number of advertisements that have hit 60 minutes. Link state advertisements are aged out after 60 minutes. Usually they will be refreshed before this time.
LS adv. flushed	Indicates number of advertisements removed (and not replaced) from the link state database.
Ptrs to Invalid LS adv	Displays number of advertisements in the database which were malformed and could not be interpreted.
Incremental ext. updates.	Displays number of changes to external destinations that are incrementally installed in the routing table.
External LSA database:	Provides information about the LSA database: **Current state** Whether the database of current AS external LSAs is in normal or overload state. **Number of LSA** The number of external LSAs currently in the database **Number of overflows** Number of times the external AS LSA database has entered overload state.

Weight

Use the **weight** command to change the cost of one of the routers OSPF interfaces. This new cost is immediately flooded throughout the OSPF routing domain, causing routes to be updated accordingly.

The cost of the interface will revert to its configured cost whenever the router is restarted or reloaded. To make the cost change permanent, you must reconfigure

the appropriate OSPF interface after invoking the weight command. This command will cause a new router links advertisement to be originated, unless the cost of the interface does not change.

Syntax:

weight *ip-interface-address new-cost*

Example: `weight 128.185.124.22 2`

Chapter 37. Using BGP4

This chapter describes how to use the Border Gateway Protocol (BGP) using the BGP configuration commands.

This chapter contains the following sections:

Border Gateway Protocol Overview

BGP is an exterior gateway routing protocol used to exchange network reachability information among autonomous systems. An AS is essentially a collection of routers and end nodes that operate under a single administrative organization. Within each AS, routers and end nodes share routing information using an interior gateway protocol. The interior gateway protocol may be either RIP or OSPF.

BGP was introduced in the Internet in the loop-free exchange of routing information between autonomous systems. Based on Classless Inter-Domain Routing (CIDR), BGP has since evolved to support the aggregation and reduction of routing information.

In essence, CIDR is a strategy designed to address the following problems:

- Exhaustion of Class B address space
- Routing table growth

CIDR eliminates the concept of address classes and provides a method for summarizing *n* different routes into single routes. This significantly reduces the amount of routing information that BGP routers must store and exchange.

Note: IBM only supports the latest version of BGP, BGP4, which is defined in RFC 1654. All references to BGP in this chapter and on the interface of IBM's routers are to BGP4, and do not apply to previous versions of BGP.

How BGP4 Works

BGP is an inter-autonomous system routing protocol. In essence, BGP routers selectively collect and advertise reachability information to and from BGP neighbors in their own and other autonomous systems. Reachability information consists of the sequences of AS numbers that form the paths to particular BGP speakers, and the list of IP networks that can be reached via each advertised path. An AS is an administrative group of networks and routers that share reachability information using one or more Interior Gateway Protocols (IGPs), such as RIP or OSPF.

Routers that run BGP are called BGP speakers. These routers function as servers with respect to their BGP neighbors (clients). Each BGP router opens a passive TCP connection on port 179, and listens for incoming connections from neighbors at this well-known address. The router also opens active TCP connections to

© Copyright IBM Corp. 1996, 1999

enabled BGP neighbors. This TCP connection enables BGP routers to share and update reachability information with neighbors in the same or other autonomous systems.

Connections between BGP speakers in the same AS are called internal BGP (IBGP) connections, while connections between BGP speakers in different autonomous systems are called external BGP (EBGP) connections.

A single AS may have one or many BGP connections to outside autonomous systems. Figure 32 shows two autonomous systems. The BGP speaker in AS1 is attempting to establish a TCP connection with its neighbor in AS2. Once this connection is established, the routers will be able to share reachability information.

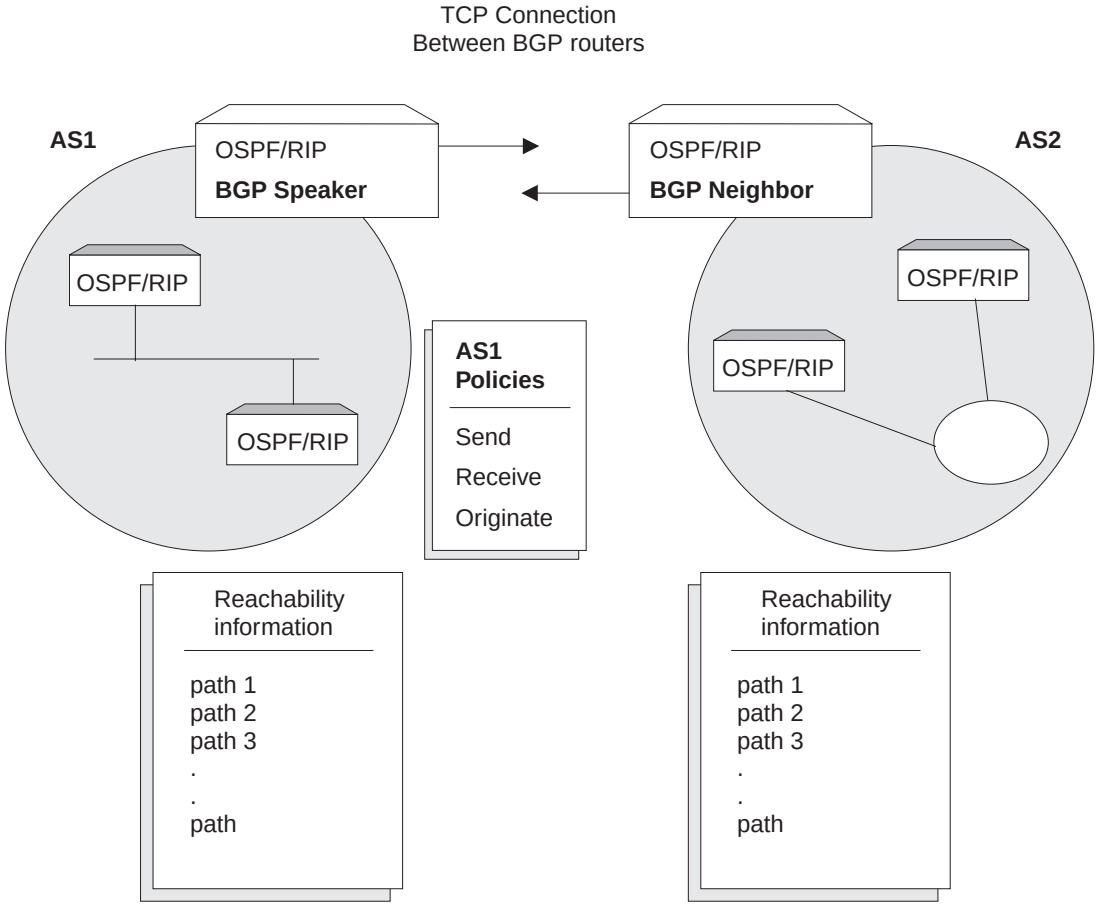

Figure 32. BGP Connections between Two Autonomous Systems. Once the BGP speaker in AS1 establishes a TCP connection with its BGP neighbor in AS2, the two routers can selectively exchange reachability information. The information each router sends or accepts is determined by policies defined for each router.

While the autonomous systems shown in Figure 32 have only one BGP router, each could have multiple connections to other autonomous systems. As an example of this, Figure 33 on page 495 shows three interconnected autonomous systems. AS1 has three BGP connections to outside autonomous systems: one to AS2, one to AS3 and one to ASx. Similarly, AS3 has connections to AS1, AS2 and to ASy.

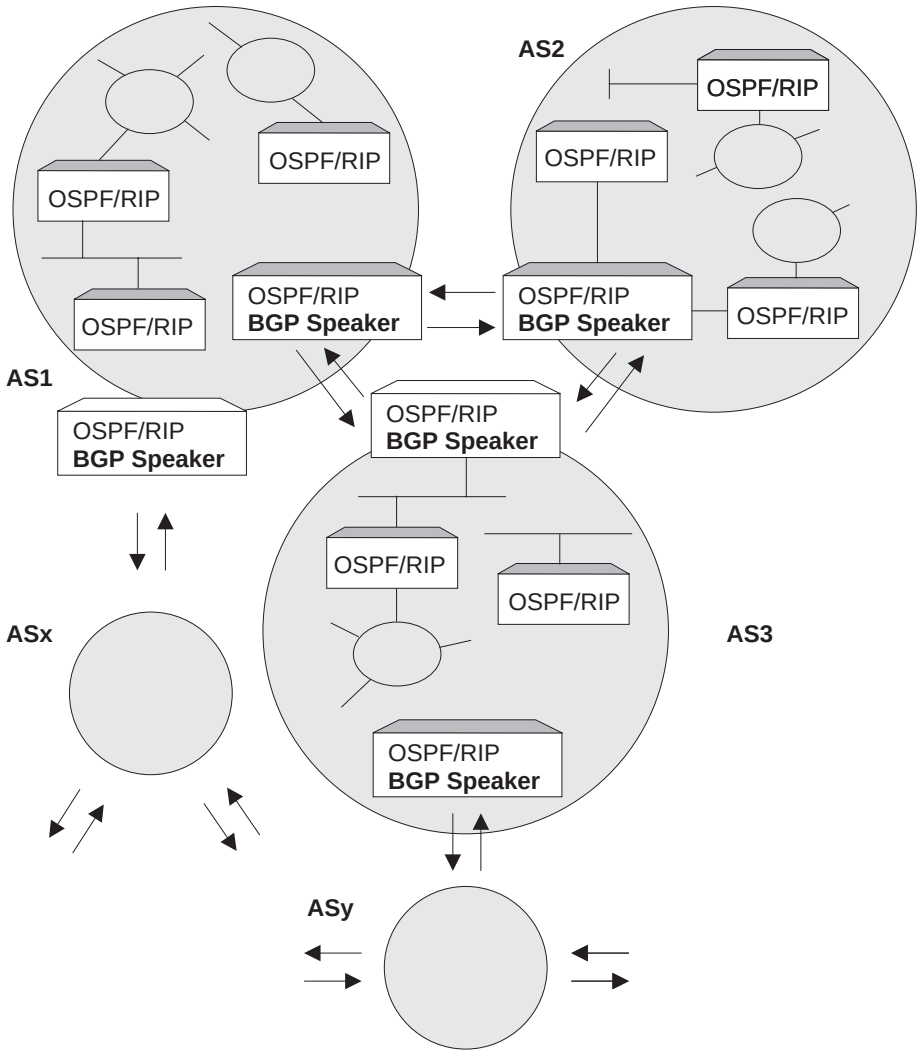

Figure 33. BGP Connections among Three Autonomous Systems. Note that AS1 and AS3 have two BGP speakers.

Once a TCP connection is established, the BGP speaker shown in Figure 32 on page 494 can send its entire routing table to its BGP neighbor in AS2. However, for security or other reasons, it may not be desirable to send reachability information on each network to AS2. Similarly, it may not be desirable for AS2 to receive reachability information on each network in AS1.

Originate, Send, and Receive Policies

Decisions on which reachability information to advertise (send), and which to accept (receive) are made on the basis of explicitly defined policy statements. IBM's BGP implementation supports three types of policy statements:

- Originate Policies
- Send Policies - There are two types of send policies
  - AS based send policies are applied only to a particular AS or all ASs. If no send policies are configured then the destination address is dropped.
  - Neighbor based send policies are applied only to a particular neighbor or neighbors. If there is no neighbor based send policies configured for a

particular neighbor, then AS based send policies are applied. If a neighbor based send policy is configured, then AS based send policy is ignored.

Each send policy statement contains the destination network advertisement classifier and a set of associated actions.

The destination network classification is based on:

- Exact destination network
- Range of destination networks
- Originating AS number
- Any AS number found in AS path attribute

The possible actions are:

- Exclude destination network for advertisement
- Include destination network for advertisement to specific AS or all ASs (using AS based policy) or to a specific neighbor (using neighbor based policy)
- Set the MED value
- ASpath padding

Note: MED and ASpath padding are only applicable to a neighbor based policy.

MED attribute value hints to external BGP neighbors about its route preference. Routes with the lowest MED attribute value will be preferred. See “Route Preference Process” on page 501 for more information.

- ASpath padding allows you to add additional local AS numbers multiple times (1 to 10) to the BGP route’s ASpath. Route with the lowest ASpath will be preferred. See “Route Preference Process” on page 501 for more information.
- Receive Policies - there are two types of receive policies.
  - AS based receive policies are applied only to a particular AS or all ASs. If no receive policies are configured then the destination address is dropped.
  - Neighbor based receive policies are applied only to a particular neighbor or neighbors. If there is no neighbor- based receive policy configured for a particular neighbor, AS based receive policies are applied. If neighbor based receive policies are configured, AS based receive policies are ignored.

Each receive policy statement contains the destination network advertisement classifier and a set of associated actions.

The destination network classification is based on:

- Exact destination network
- Range of destination networks
- Originating AS number
- Any AS number found in AS path attribute

The possible actions are:

- Exclude destination network
- Include destination network from a specific AS or all ASs (using AS based policy) or from a specific neighbor (using neighbor based policy)
- Reset the MED value
- Set weight value

- Set IGP metric value
- Set local preferences value.

Note: MED, weight, and local preferences are only applicable to a neighbor based policy.

Weight value hints to local BGP routers to select the route based on highest weight value and ignores the route preference algorithm.

BGP Messages

BGP routers use four kinds of messages to communicate with their neighbors: OPEN, KEEP ALIVE, UPDATE, and NOTIFICATION messages.

OPEN

Open messages are the first messages transmitted when a link to a BGP neighbor comes up and establishes a connection.

KEEP ALIVE

Keep alive messages are used by BGP routers to inform one another that a particular connection is alive and working.

UPDATE

Update messages contain the interior routing table information. BGP speakers send update messages only when there is a change in their routing tables.

NOTIFICATION

Notification messages are sent whenever a BGP speaker detects a condition that forces it to terminate an existing connection. These messages are advertised before the connection is transmitted.

Setting Up BGP4

Setting up BGP involves three basic steps:

1. “Enabling BGP”.

   Enabling BGP requires you to specify the BGP router’s unique AS Number. AS numbers are assigned by Stanford Research Institute Network Information Center.
2. “Defining BGP Neighbors” on page 498.

   BGP Neighbors are BGP routers with which a BGP speaker establishes a TCP connection. Once neighbors are defined, connections to them are established by default.
3. “Adding Policies” on page 498.

   The *policies* you establish determine which routes will be imported and exported by the BGP speaker. You can set up policies for different purposes. See “Sample Policy Definitions” on page 498 for more information.

Enabling BGP

You enable BGP using the **enable BGP speaker** command as shown.

```
BGP Config> enable BGP speaker
AS [0]? 167
TCP segment size [1024]?
```

The *AS number* must be in the range 1 to 65535. The *TCP segment* size must be in the range 1 to 65535. The default value for *TCP segment* is 1024. This number represents the maximum segment size BGP will use for passive TCP connections.

After you have issued the **enable bgp** command you must reboot the device to enable BGP.

Defining BGP Neighbors

After enabling a BGP speaker, you must define its neighbors. BGP neighbors can be internal or external. Internal neighbors exist in the same AS and do not need to have a direct connection to one another. External neighbors exist in different autonomous systems. These must have a direct connection to one another.

To define internal or external BGP neighbors, use the **add neighbor** command. You must specify the IP address of the neighbor, and assign an AS number to the neighbor as shown below. Internal neighbors must have the same AS number as the BGP speaker.

```
BGP Config> add neighbor 192.0.190.178
AS [0]? 178
Init timer [12]? 30
Connect timer [120]?
Hold timer [90]? 30
TCP segment size [1024]? 512
```

Use the **reset neighbor** command to activate the specified BGP neighbor, based on the neighbor configuration parameters stored in the configuration memory.

Adding Policies

IBM's BGP implementation supports three policy commands:

- *Originate Policy*. This enables you to select the interior gateway protocol (IGP) networks to export.
- *Receive Policy*. This enables you to select the route information to import from BGP peers.
- *Send Policy*. This enables you to select the route information to export to peers. Note that exportable route information can include information collected from neighboring autonomous systems, as well as the routes that originate in the IGP.

If you added or modified a neighbor based policy use the **reset neighbor** command to activate the neighbor policy. If you added or modified an AS-based policy you must reboot the device.

Sample Policy Definitions

This section provides a set of examples of some specific policies you can set up for a BGP speaker. All policies are defined using the BGP **add** command. See "Add" on page 504 for the syntax of the **add** command.

Originate Policy Examples

Include All Routes for Advertisement

This example includes all routes in the BGP speaker's IGP routing table for advertisement. In this sense, you can view this command as the "default" originate policy statement for BGP.

Notice that the command specifies a range of addresses, rather than a single (exact) address.

```
BGP Config> add originate-policy inclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Tag [0]?
```

Exclude a Range of Routes

This example also specifies a range, but in this case the goal is to prevent the BGP Speaker from advertising addresses in this range to its neighbors.

This example excludes all routes in the range 194.10.16.0 to 194.10.31.255 from the IGP routing table, which in turn prevents them from being advertised.

```
BGP Config> add originate-policy exclusive
Network Prefix [0.0.0.0]? 194.10.16.0
Network Mask [0.0.0.0]? 255.255.240.0
Address Match (Exact/Range) [Exact]? range
Tag [0]?
```

The tag is the received RIP information. You can select networks based on a particular tag value for advertisement. See the description of the **Set** command in "Chapter 30. Configuring and Monitoring IP" on page 307 for information on setting the tag value.

By default, only classfull routes from the BGP speaker's IGP routing table will be selected for advertisement. To select a classless route for advertisement use the bgp-subnets **patch** command. See "Patch" on page 65.

AS Based Receive Policy Examples

Import all Routes from all BGP Neighbors

This example ensures that the BGP speaker will import all routes from all of its neighbors into its IGP routing table.

```
BGP Config> add receive-policy inclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Originating AS# [0]?
Adjacent AS# [0]?
IGP-metric [0]?
```

IGP-metric specifies the metric value with which the accepted routes are imported into the speaker's IGP routing table. You are only prompted to enter a value for IGP-metric only when setting up a policy for route inclusion.

If *IGP-metric* is -1, these routes will not be imported into IGP; thus, routes are not re-advertisable.

Block Specific Routes from an Originating AS

This example will prevent the BGP speaker from importing any routes originating at AS 168 from neighboring AS 165. You might use this command if you do not want the BGP speaker to receive any routes from AS 168 for security reasons.

```
BGP Config> add receive-policy exclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Originating AS# [0]? 168
Adjacent AS# [0]? 165
```

Block Specific ASpath

This example will prevent the BGP speaker from importing any route that has AS 175 in its ASpath list.

```
BGP Config> add no-receive
Enter AS: [0]? 175
```

Neighbor Based Receive Policy Examples

Import all routes from a specific BGP neighbor, set weight = 100

This example will allow you to import all routes from BGP neighbor 192.0.190.178. All routes will have a weight value of 100 and IGP-metric value of 1.

Define the policy list name for receive policy.

```
BGP Config> add policy-list
Name[]?S1_100_r
Policy Type(Receive/Send)[Receive]?Receive
```

Attach the defined receive policy list name to a specific neighbor.

```
BGP Config> attach policy-to-neighbor
Neighbor address [0.0.0.0]?192.0.190.178
First receive policy list name (none for global AS based policy)[]?S1_100_r
Second receive policy list name (none for exit)[]?
```

Add receive policies for neighbor using **update** and **add** command.

```
BGP Config>update policy S1_100_r
Policy-list S1_100_r Config>add
Policy type (Inclusive/Exclusive) [Exclusive]? inclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Range]?
Originating AS# [0]?
Any AS# [0]?
MED [0]?
Weight [0]? 100
Local-Pref [0]?
IGP-metric [0]? 1
```

AS Based Send Policy Examples

Restrict Route Advertisement to a Specific AS

This example restricts the BGP speaker. The speaker cannot advertise routes in the address range 143.116.0.0 to 143.116.255.255, that originate from AS 165, to autonomous system 168.

```
BGP Config> add send exclusive
Network Prefix [0.0.0.0]? 143.116.0.0
Network Mask [0.0.0.0]? 255.255.0.0
Address Match (Exact/Range) [Exact]? range
Tag [0]? 165
Adjacent AS# [0]? 168
```

Advertise All Known Routes

This example ensures that the BGP speaker will advertise all routes originated from its IGP, and all routes learned from its neighboring autonomous systems.

```
BGP Config> add send policy inclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Tag [0]?
Adjacent AS# [0]?
```

Neighbor Based Send Policy Examples

Advertise All Known Routes to a Specific Neighbor with MED Attribute value = 100

This example will allow you to advertise all routes to a BGP neighbor 192.0.190.178. All advertise routes will have a MED value of 100.

Define the policy list name for send policy.

```
BGP Config> add policy-list
Name[]?S1_100_s
Policy Type(Receive/Send)[Receive]?Send
```

Attach the defined send policy list name(s) to a specific neighbor.

```
BGP Config> attach policy-to-neighbor
Neighbor address [0.0.0.0]?192.0.190.178
First send policy list name (none for global AS based policy)[]?S1_100_s
Second send policy list name (none for exit)[]?
```

Add the send policies for neighbor using the **update** and **add** commands.

```
BGP Config>update policy S1_100_s
Policy-list S1_100_s Config>add
Policy type (Inclusive/Exclusive) [Exclusive]?
Network prefix [0.0.0.0]?
Network mask [0.0.0.0]?
Address match (exact/range) [range]?
Originating AS# [0]?
TAG [0]?
MED [0]? 100
# of AS to pad [0]?
```

Route Preference Process

When the BGP speaker receives a path for particular destination from its peer, BGP goes through the following process for selecting a best possible path:

- Applies receiving policies based on configuration.
- If a destination is permitted by receiving policies, then it calculates Degree of Preference for the received destination, based on shorter ASpath length and Origin type.
- If there are several paths to the same destination then, it executes the path selection process. It selects the best possible path by comparing the new path with the existing selected best path. If the new path is selected as the best path, then it installs the new path in the IP forwarding table.

- BGP advertises the selected best path to its External and Internal BGP peers, subject to send policies.

Path Selection Process

The best path is selected based on the following order:

- Prefer the path that has been originated by this router.
- If path is not originated by this router, then prefer the path which has highest configured Weight value.
- If paths have same weight value then, prefer the path which has highest configured local-preference value.
- If paths have same local-preference value, then prefer the path which has highest Degree of Preference.
  - The path which has shortest ASpath length is given higher degree of preference.
  - If paths have same ASpath length, then Origin type IGP is preferred over EGP and Incomplete.
- If paths have same Degree of Preference, then prefer the path which has the lowest MED attribute value.
- If paths have same MED attribute value, then prefer External(EBGP) over internal (IBGP) route.
- If paths are still same, then prefer the path with lowest BGP-ID.

Chapter 38. Configuring and Monitoring BGP4

This chapter describes the BGP configuring and monitoring commands and includes the following sections:

- "BGP4 Configuration Commands"
- "Accessing the BGP4 Configuration Environment"
- "Accessing the BGP Monitoring Environment" on page 517
- "BGP4 Monitoring Commands" on page 517

Accessing the BGP4 Configuration Environment

To access the BGP configuration environment, enter the following command at the `Config>` prompt:

```
Config> Protocol BGP
BGP Config>
```

BGP4 Configuration Commands

This section describes the BGP configuration commands. These commands allow you to modify the BGP protocol behavior to meet your specific requirements. Some amount of configuration is necessary to produce a fully functional BGP router. Enter BGP configuration commands at the `BGP config>` prompt.

Table 69. BGP Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Add BGP neighbors and policies.
Attach	Attaches receive and send policy-list to a particular neighbor.
Change	Modifies information that was originally entered with the **add** command.
Delete	Deletes BGP configuration information that had been entered with the **add** command.
Disable	Disables certain BGP features that have been turned on by the **enable** command.
Enable	Enables BGP speakers, BGP neighbors or Classless BGP.
List	Displays BGP configuration items.
Move	Changes the order in which policies and aggregates are defined.
Set	Sets the IP-route-table-scan-timer.
Update	Manipulates a policy in a configured policy-list name using the submenu **add**, **delete**, **change** and **move** commands.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

© Copyright IBM Corp. 1996, 1999

Add

Use the **add** command to add BGP information to your configuration.

Syntax:

add aggregate . . .
neighbor . . .
no-receive asnum . . .
originate-policy . . .
policy-list . . .
receive-policy . . .
send-policy. . .

aggregate *network prefix network mask*

The **add aggregate** command causes the BGP speaker to aggregate a block of addresses, and advertise a single route to its BGP neighbors. You must specify the network prefix common to all the routes being aggregated and its mask. The following example illustrates how to aggregate a block of addresses from 194.10.16.0 through 194.10.31.255.

1. The *Network Prefix* is the addresses being affected. The prefix is the first address in a range of addresses specified in a BGP policy.
   Valid Values: Any valid IP address.
   Default Value: none
2. The *Network Mask* applies to the address specified in Network Prefix to generate an address used in a BGP policy.
   Valid Values: Any valid IP address.
   Default Value: none

Example:

```
add aggregate

Network Prefix [0.0.0.0]? 194.10.16.0
Network Mask [0.0.0.0]? 255.255.240.0
```

When you add an aggregate definition, remember to define a policy to block the aggregated routes from being exported. If you do not, the router will advertise both the individual routes and the aggregate you have defined. This does not apply when you are aggregating the routes, which are originated from it's IGP routing table.

neighbor *neighbor IP address as# init timer connect timer hold timer keep alive timer tcp segment size*

Use the **add neighbor** command to define a BGP neighbor. The neighbor can be internal to the BGP speaker's AS, or external.

1. The IP address is the address of the neighbor you wish to peer with. It could be within your own autonomous system or in another autonomous system. If it is an external neighbor, both BGP speakers must share the same network. There is no such restriction for internal neighbors. The address has:
   Valid Values: Any valid IP address.
   Default Value: none
2. The AS number is your own autonomous system number for internal neighbor or neighbor's autonomous system number. The AS number of the neighbor has:
   Valid Values: An integer in the range of 0 - 65535
   Default Value: none

3. The *Init timer* to specifies the amount of time the BGP speaker waits to initialize resources and reinitiate transport connection with the neighbor in case the speaker has previously changed to IDLE state due to an error. If the error persists, this timer increases exponentially.
   Valid Values: 0 to 65535 seconds.
   Default Value: 12 seconds
4. The *Connect timer* specifies the amount of time the BGP speaker waits to reinitiate transport connection to its neighbor, if the TCP connection fails while in either CONNECT or ACTIVE state. In the mean time, the BGP speaker continues to listen for any connection that may be initiated by its neighbor.
   Valid Values: 0 to 65535 seconds.
   Default Value: 120 seconds
5. Enter the *Hold timer* to specify the length of time the BGP speaker waits before assuming that the neighbor is unreachable. Both neighbors exchange the configured information in OPEN message and choose the smaller of the two timers as their negotiated Hold Timer value.

   Once neighbors have established BGP connection, they exchange Keepalive messages at frequent intervals to ensure that the connection is still alive and the neighbors are reachable. The Keep-Alive timer interval is calculated to be one-third of the negotiated hold timer value. Hence the hold timer value must be either zero or at least three seconds.

   Note that on switched lines, you may wish to have the Hold Timer value of zero to save bandwidth by not sending Keepalives at frequent intervals.
   Valid Values: 0 to 65535 seconds.
   Default Value: 90 seconds
6. The *TCP segment size* specifies the maximum data size that may be exchanged on the TCP connection with a neighbor. This value is used for active TCP connection with the neighbor.
   Valid Values: 0 to 65535 bytes.
   Default Value: 1024 bytes

Example:

```
        add neighbor

Neighbor address [0.0.0.0]? 192.0.251.165
AS [0]? 165
Init timer [12]?
Connect timer [120]?
Hold timer [90]?
TCP segment size [1024]?
```

no-receive *asnum*

Use the **add no-receive asnum** to exclude AS-paths if the particular AS number appears anywhere inside the AS-path list.

The *AS number* has:
Valid Values: 0 to 65535
Default Value: none

Example:

```
        add no-receive

Enter AS: [0]? 178
```

originate-policy *(exclusive/ inclusive) network prefix network mask address match (Exact/Range) tag*

Use the **add originate-policy** command to create a policy that determines whether a specific address, or range of addresses, can be imported to the BGP speaker's routing table from the IGP routing table.

Exclusive
Exclusive policies prevent route information from being included in the BGP speaker's routing table.

Inclusive
Inclusive policies ensure that specific routes will be included in the BGP speaker's routing table.

Network prefix
The network prefix for the addresses being affected.
Valid Values: Any valid IP address.
Default Value: none

Address match
The address, or range of addresses, that will be affected by the policy statement.Enter the *Network Mask* to be applied to the address specified in Network Prefix to generate an address used in a BGP policy.
Valid Values: Any valid IP address.
Default Value: none

Tag The value that has been set for a particular AS. All tag values match that of the AS from which they were learned.
Valid Values: 0 to 65535
Default Value: none

The following example includes all routes in the BGP speaker's IGP routing table to be advertised.

Example:

```
add originate-policy exclusive

Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Exact]? range
Tag [0]?
```

See "Originate Policy Examples" on page 499 for detailed examples of this policy command.

policy-list
Use the **add policy-list** command to configure a group of policy, which can be attached to a specific neighbor using the **attach policy-to-neighbor** command.

Example: add policy-list

```
Name[]? nbr1-rcv
Policy Type(Receive/Send)[Receive]?Receive
```

Example: add policy-list

```
Name[]? nbr1-snd
Policy Type(Receive/Send)[Receive]?Send
```

Note: See "Neighbor Based Receive Policy Examples" on page 500 and "Neighbor Based Send Policy Examples" on page 501 for detailed examples of this policy command.

receive-policy *(exclusive/ inclusive) network prefix network mask address match originating as# adjacent as# igpmetric (inclusive only)*

Use the **add receive-policy** command to determine what routes will be imported to the BGP speaker's routing table.

Exclusive policies prevent route information from being included in the BGP speaker's routing table.

1. The *Network Prefix* is the addresses being affected.
   Valid Values: Any valid IP address.
   Default Value: none
2. The *Network Mask* applies to the address specified in Network Prefix to generate an address used in a BGP policy.
   Valid Values: Any valid IP mask.
   Default Value: none
3. The *Address match* is a range of addresses or an exact address.
4. An *Originating AS#* has:
   Valid Values: 0 to 65535
   Default Value: none
5. The *Adjacent AS#* to specifies the neighboring AS number.
   Valid Values: 0 to 65535
   Default Value: none

Example:

```
add receive-policy exclusive

Network Prefix [0.0.0.0]? 10.0.0.0
Network Mask [0.0.0.0]?  255.0.0.0
Address Match (Exact/Range) [Exact]? range
Originating AS# [0]? 168
Adjacent AS# [0]? 165
```

See "AS Based Receive Policy Examples" on page 499 for detailed examples of this policy command.

send-policy *(exclusive/ inclusive) network prefix network mask address match tag adjacent as#*

Use the **add send-policy** command to create policies that determine which of the BGP speaker's learned routes will be readvertised. These routes could be internal or external to the BGP speaker's AS.

Exclusive policies prevent route information from being included in the BGP speaker's routing table.

1. The *Network Prefix* is for the addresses being affected.
   Valid Values: Any valid IP address.
   Default Value: none
2. The *Network Mask* applies to the address specified in Network Prefix to generate an address used in a BGP policy.
   Valid Values: Any valid IP address.
   Default Value: none
3. The *Address match* is a range of addresses or an exact address.
4. A *TAG*. is the value that has been set for a particular AS. Tag values match that of the AS from which they were learned.
   Valid Values: 0 to 65535
   Default Value: none
5. The *Adjacent AS#* specifies the neighboring AS number.
   Valid Values: 0 to 65535
   Default Value: none

Example:

```
add send exclusive
```

```
Network Prefix [0.0.0.0]? 180.220.0.0
Network Mask [0.0.0.0]? 255.255.0.0
Address Match (Exact/Range) [Exact]? range
Tag [0]?
Adjacent AS# [0]? 25
```

See “AS Based Send Policy Examples” on page 500 for detailed examples of this policy command.

Attach

Use the **attach policy-to-neighbor** command to attach a configured policy-list name to a specific neighbor. You can attach up to three receive and three send policy-list names.

Syntax:

attach	policy-to-neighbor

Example: attach policy-to-neighbor

```
Neighbor address [0.0.0.0]? 192.0.251.165
First receive policy list name (none for global AS based policy)[]? nbr1-rcv
Second receive policy list name (none for exit)[]?
First send policy list name (none for global AS based policy)[]? nbr1-snd
Second send policy list name (none for exit)[]?
```

Note: See “Neighbor Based Receive Policy Examples” on page 500 and “Neighbor Based Send Policy Examples” on page 501 for detailed examples of this policy command.

Change

Use the **change** command to change a BGP configuration item previously installed by the add command.

Syntax:

change	aggregate . . .
	neighbor . . .
	originate-policy . . .
	policy-to-neighbor
	receive-policy . . .
	send-policy. . .

aggregate *index# network prefix network mask*

This example changes the current aggregate (aggregate 1). The change causes aggregate 1 to use a different network prefix and mask to aggregate all routes in the address range from 128.185.0.0 to 128.185.255.255.

Example:

```
change aggregate 1

Network Prefix [128.185.0.0]? 128.128.0.0
Network Mask [255.255.0.0]? 255.192.0.0
```

neighbor *neighbor IP address as# init timer connect timer hold timer keep alive timer tcp segment size*

The following example changes the value of the hold timer to zero for neighbor 192.0.251.165.

The *neighbor address* to be modified has:
Valid Values: Any valid IP address.
Default Value: none
Example:

```
change neighbor 192.0.251.165

AS [165]?
Init timer [12]?
Connect timer [60]?
Hold timer [12]? 0
TCP segment size [1024]?
```

originate-policy *index# (exclusive/ inclusive) network prefix network mask address match tag*

Use the **change originate-policy** command to alter an existing originate policy definition.

This example alters the BGP speaker's originate policy. Rather than excluding networks with prefix 194.10.16.0 from the IGP routing table, the policy will now include all routes.
Example:

```
change originate-policy

Enter index of originate-policy to be modified [1]?
Policy Type (Inclusive/Exclusive) [Exclusive]? inclusive
Network Prefix [194.10.16.0]? 0.0.0.0
Network Mask [255.255.240.0]? 0.0.0.0
Address Match (Exact/Range) [Range]?
Tag [0]?
```

policy-to-neighbor

Use the **change policy-to-neighbor** command to change a policy-list attachment to a particular neighbor.
Example:

```
change policy-to-neighbor

Neighbor address [0.0.0.0]? 192.0.251.165
First receive policy list name to be changed[nbr1-rcv]?
Second receive policy list name to be changed[]?
Third receive policy list name to be changed[]?
First send policy list name to be changed[nbr1-snd]?
Second send policy list name to be changed[]?
Third send policy list name to be changed[]?
```

receive-policy *index# (exclusive/inclusive) network prefix network mask address match originating as# adjacent as# igpmetric (inclusive only)*

Use the **change receive-policy** command to alter an existing receive policy definition.

This example adds a restriction to the BGP speaker's receive-policy. Rather than import route information from every BGP peer into its IGP routing table, it will now prevent routes from AS 165 from being imported.
Example:

```
change receive-policy

Enter index of receive-policy to be modified [1]?
Policy Type (Inclusive/Exclusive) [Inclusive]? exclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Range]?
Originating AS# [0]?
Adjacent AS# [0]? 165
```

send-policy *index# (exclusive/ inclusive) network prefix network mask address match tag adjacent as#*

Use the **change send-policy** command to alter an existing send policy to one that is more inclusive, or more exclusive.

This example adds a restriction to the BGP speaker's send policy. The restriction ensures that all routes in the address range 194.10.16.0 to 194.10.31.255 will be excluded when advertising to autonomous system 165.

Example:

```
        change send-policy

Enter index of send-policy to be modified [1]?
Policy Type (Inclusive/Exclusive) [Inclusive]? exclusive
Network Prefix [0.0.0.0]? 194.10.16.0
Network Mask [0.0.0.0]? 255.255.240.0
Address Match (Exact/Range) [Range]?
Tag [0]?
Adjacent AS# [0]? 165
```

Delete

Use the **delete** command to delete a BGP configuration item previously installed by the **add** command.

Syntax:

delete aggregate . . .
neighbor . . .
no-receive . . .
originate-policy . . .
policy-list . . .
policy-to-neighbor
receive-policy . . .
send-policy. . .

aggregate *index#*
You must specify the index number of the aggregate you want to delete.

Example: `delete aggregate 1`

neighbor *neighbor IP address*
Use this command to delete a BGP neighbor. You must specify the neighbor's network address.

The *neighbor's network address to be deleted* has:
Valid Values: Any valid IP address.
Default Value: none

Example: `delete neighbor 192.0.251.165`

no-receive *as*
Use this command to delete the no-receive policy set up for a particular AS. You must specify the AS number.

The *AS number* has:
Valid Values: 0 to 65535
Default Value: none

Example: `delete no-receive 168`

originate-policy *index#*
Use this command to delete a specific originate policy. You must specify the index number associated with the policy.

Example: `delete originate-policy 2`

policy-list
Use the **delete policy-list** command to delete a policy-list.

Example: `delete policy-list`

```
Name of policy-list to delete []? nbr1-rcv
All policies defined for 'nbr1-rcv' will be deleted.
Are you sure you want to delete (Yes or [No]? Yes
Policy-list 'nbr1-rcv' is deleted.
```

The policy-to-neighbor attachment will be adjusted accordingly.

policy-to-neighbor

Use the **delete policy-to-neighbor** command to delete an existing policy-list name attachment to a particular neighbor.

Example: `delete policy-to-neighbor`

```
Neighbor address [192.0.251.165]?
Remove first receive policy-list name [nbr1-rcv]
Are you sure you want to remove (Yes or [No])? yes
Remove first send policy-list name [nbr1-snd]
Are you sure you want to remove (Yes or [No])? yes
```

receive-policy *index#*

Use this command to delete a specific receive policy. You must specify the index number associated with the policy.

Example: `delete receive-policy`

```
Enter index of receive-policy to be deleted [1]?
```

send-policy *index#*

Use this command to delete a specific send policy. You must specify the index number associated with the policy.

Example: `delete send-policy 4`

Disable

Use the **disable** command to disable a previously enabled BGP neighbor or speaker. Note that neighbors are implicitly enabled whenever added with the **add** command.

Syntax:

disable	BGP speaker
	classless-bgp
	compare-med-from-diff-AS
	neighbor . . .

bgp speaker

Use the **disable bgp speaker** command to disable the BGP protocol.

Example: `disable bgp speaker`

classless-bgp

Use this command to disable a classless route for advertisement.

Example: `disable classless-bgp`

Note: Be sure that the **patch bgp-subnets** command is disabled.

compare-med-from-diff-AS

Use this command to disable a MED comparison between different ASs.

Example: `disable compare-med-from-diff-AS`

neighbor *neighbor IP address*

The *neighbor address* has:

Valid Values: Any valid IP address.
Default Value: none

Example: `disable neighbor 192.0.190.178`

Enable

Use the **enable** command to activate the BGP features, capabilities, and information added to your BGP configuration.

Syntax:

enable BGP speaker
classless-bgp
compare-med-from-diff-AS
neighbor . . .

bgp speaker *as# tcp segment size*

Use the **enable bgp speaker** command to enable the BGP protocol.

Note: IBM only supports the latest version of BGP - BGP4, which is defined in RFC 1654.

1. The *AS number* is associated with this collection of routers and nodes.
   Valid Values: 0 to 65535
   Default Value: none
2. Enter the *TCP segment size* to specify the maximum segment size that BGP should use for passive TCP connections.
   Valid Values: 0 to 65535 bytes.
   Default Value: 1024 bytes

Example:

```
enable bgp speaker

AS [0]? 165
TCP segment size [1024]?
```

classless-bgp neighbor

Use this command to enable a classless route for advertisement.

Example: `enable classless-bgp`

compare-med-from-diff-AS

Use this command to enable MED comparison between different ASs.

Example: `enable compare-med-from-diff-AS`

neighbor *neighbor IP address*

Use this command to enable a BGP neighbor.

The *neighbor address* has:
Valid Values: Any valid IP address.
Default Value: none

Example: `enable neighbor 192.0.190.178`

List

Use the **list** command to display various pieces of the BGP configuration data, depending on the particular subcommand invoked.

Syntax:

list aggregate
all
BGP speaker
neighbor
no-receive
originate-policy
policy-list . . .
policy-to-neighbor
receive-policy
send-policy

aggregate

Use the **list aggregate** command to all aggregated routes defined with the **add aggregate** command.

Example: `list aggregate`

```
Aggregation:
Index    Prefix              Mask
1         194.10.16.0        255.255.240.0
```

all Use the **list all** command to list the BGP neighbors, policies, aggregated routes, and no-receive-as records in the current BGP configuration.

Example: `list all`

```
                  BGP Protocol:          Enabled
                  AS:                    167
                  TCP-Segment Size:      1024
                  Neighbors and their AS:

                                         Init    Conn    Hold    TCPSEG
Address           State     AS           Timer   Timer   Timer   Size
128.185.250.168   ENABLD    168    12      60      12      1024
192.0.251.165     ENABLD    165    12      60      12      1024

Receive-Policies:
Index  Type  Prefix       Mask      Match  OrgAS  AdjAS IGPmetric
1      INCL  0.0.0.0      0.0.0.0   Range  0      0      0

Send-Policies:
Index  Type  Prefix       Mask      Match  Tag    AdjAS
1      INCL  0.0.0.0      0.0.0.0   Range  0      0

Originate-Policies:
Index  Type  Prefix       Mask             Match  Tag
1      EXCL  194.10.16.0  255.255.240.0    Range  0

Aggregation:

Index  Prefix             Mask
1      194.10.16.0        255.255.240.0
No no-receive-AS records in configuration.
```

bgp speaker

Use the **list bgp speaker** command to derive information on the BGP speaker. The information provided is as follows:

Example:

`list BGP speaker`

```
BGP Protocol:          Enabled
AS:                    165
TCP-Segment Size:      1024
```

neighbor

Use the **list neighbor** command to derive information on BGP neighbors.

Example: `list neighbor`

```
Neighbors and their AS:

                                        Init    Conn    Hold    TCPSEG
Address            State     AS         Timer   Timer   Timer   Size

128.185.252.168    ENABLD    168    12      60      12      1024
192.0.190.178      DISBLD    178    12      60      12      1024
192.0.251.167      ENABLD    167    12      60      12      1024
```

no-receive

Use the **list no-receive** command to derive information on no-receive-AS definitions that have been added to the BGP configuration.

Example: `list no-receive`

```
AS-PATH with following autonomous systems will be discarded:
AS  178
AS  165
```

originate-policy *all index prefix*

Use the **list originate-policy** command to derive information on the originate policies that have been added to the BGP configuration.

Example: `list originate-policy`

```
Originate-Policies:
Index  Type  Prefix          Mask            Match  Tag
1      EXCL  194.10.16.0     255.255.240.0   Range  0
2      INCL  0.0.0.0         0.0.0.0         Range  0
```

policy-list

Use the **list policy-list** command to list configured policy-list names.

Example: `list policy-list`

```
BGP Config>li policy list
Policy list:
nbr1-rcv  Receive
nbr1-snd  Send
```

policy-to-neighbor

Use the **list policy-to-neighbor** command to list policies attached to neighbors.

Example: `list policy-to-neighbor`

```
Neighbor addrs  receive         send
192.0.251.165   nbr1-rcv        nbr1-snd
```

receive-policy adj-as-number *all* **or** *index* **or** *prefix*

Use the **list receive-policy** command to derive information on the receive policies that have been added to the BGP configuration. You can display all receive policies defined for an AS, or display policies by index or prefix number.

Example: `list receive-policy`

```
Receive-Policies:
Index  Type  Prefix      Mask        Match  OrgAS  AdjAS  IGPmetric
1      EXCL  0.0.0.0     0.0.0.0     Range  178    165
2      INCL  0.0.0.0     0.0.0.0     Range  0      0      0
```

send-policy adj-as-number *all* **or** *index* **or** *prefix*

Use the **list send-policy** command to display information on send policies defined for specified autonomous systems. You can display all send policies defined for an AS, or display policies by index or prefix number.

Example: `list send-policy`

```
Send-Policies:
Index  Type  Prefix      Mask           Match Tag    AdjAS
1      EXCL  194.10.16.0 255.255.240.0  Range   0    165
2      INCL   0.0.0.0    0.0.0.0        Range   0    0
```

Move

Use the **move** command to change the order in which policies and aggregates have been defined. This changes the order in which the router applies existing policies to route information. Before using this command, it is advisable to use the **list** command to see what policies have been defined.

Syntax:

move	*aggregate* or *originate-policy* or *receive-policy* or *send-policy*

Example:

```
move originate-policy
Enter index of originate-policy to move [1]? 3
Move record AFTER record number [0]?
```

Set

Use the **set** command to set the IP-route-table-scan-timer. The IP-route-table-scan-timer value is used to set the IP forwarding table scanning time interval for BGP updates.

Syntax:

set	ip-route-table-scan-timer

Example:

```
set ip-route-table-scan-timer
```

Update

Use the **update** command and sub-commands to manipulate policies.

Syntax:

update	*policy-list*

Receive Policy Example:

```
update policy-list
Name[]? nbr1-rcv
```

Add

Use the **Add** command to add receive policies within the **update** command.

```
BGP nbr1-rcv: Receive Config>add
Policy type (Inclusive/Exclusive) [Exclusive]? inclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Range]?
Originating AS# [0]?
Any AS# [0]?
MED [0]?
Weight [0]?
Local-Pref [0]?
IGP-metric [0]?
```

Note: There will be no prompting for MED, Local-pref, Weight, and IGP-metric parameters for exclusive receive policy. MED and Local-pref values will be

used from received advertisement if they are configured as value '0'. The value '0' for the weight parameter indicates to ignore the weight value in the route selection process.

Change

Use the **Change** command to change policies within the **update** command.

Example:

```
Enter index of receive-policy to be modified [1]?
```

Delete

Use the **delete** command to delete policies within the **update** command.

Example:

```
Enter index of receive-policy to be deleted [1]?
```

Move

Use the **move** command to move policies within the **update** command.

Example:

```
Enter index of receive-policy to move [1]?
Move record after record number [0]?
```

List

Use the **list policy-list** command to list receive policies within the **update** command.

Example: list policy-list

```
Receive policy list for 'name':
                    T Prefix            Match OrgAS AnyAS MED   Weight Lpref IGPmetric
                1   I 0.0.0.0/0         Range 0     0     0     0      0     1
```

Send Policy Example:

```
update policy-list
Name[]? nbr1-rcv

```

Add

Use the **Add** command to add send policies within the **update** command.

```
BGP nbr1-rcv: Send Config>add
Policy type (Inclusive/Exclusive) [Exclusive]? inclusive
Network Prefix [0.0.0.0]?
Network Mask [0.0.0.0]?
Address Match (Exact/Range) [Range]?
Originating AS# [0]?
Any AS# [0]?
TAG [0]
MED [0]?
# of AS to pad[0]?
```

Note: There will be no prompting for MED and ASpad parameters for exclusive send policy. The value 0 for the MED parameter indicates that MED attribute is not included in advertisement. The value 0 for the ASpad parameter indicates that there will be no additional local AS number inserted in the ASpath.

Change

Use the **Change** command to change policies within the **update** command.

Example:

```
Enter index of send-policy to be modified [1]?
```

Delete

Use the **delete** command to delete policies within the **update** command.

Example:

```
Enter index of send-policy to be deleted [1]?
```

Move

Use the **move** command to move policies within the **update** command.

Example:

```
Enter index of send-policy to move [1]?
Move record after record number [0]?
```

List

Use the **list policy-list** command to list send policies within the **update** command.

Example: list policy-list

```
Send policy list for 'name':
                    T Prefix            Match OrgAS AnyAS Tag   MED    ASpad
                1   I 0.0.0.0/0         Range 0     0     0     0      0
```

Accessing the BGP Monitoring Environment

To access the BGP monitoring environment, enter the following command at the + prompt:

```
+ Protocol BGP
BGP>
```

BGP4 Monitoring Commands

This section describes the BGP monitoring commands. These commands allow you to modify the BGP protocol behavior to meet your specific requirements. Some amount of configuration is necessary to produce a fully functional BGP router. Enter BGP monitoring commands at the BGP> monitoring prompt.

Table 70. BGP Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Destinations	Displays all entries in the BGP routing table.
Dump routing tables	Lists the contents of the IP routing table.
Neighbors	Displays currently active neighbors.
Parameter	Displays installed BGP globals in the BGP system.
Paths	Displays all available paths in the database.

Table 70. BGP Monitoring Command Summary (continued)

Command	Function
Ping	Sends ICMP Echo Requests to another host once a second and watch for a response. This command can be used to isolate trouble in an internetwork environment.
Policy-list	Displays the current installed policy for specific neighbor and usage statics of each policy.
Reset neighbor	Resets a particular neighbor.
Traceroute	Displays the complete path (hop-by-hop) to a particular destination.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Destinations

Use the **destinations** command to dump all BGP routing table entries, or to display information on routes advertised to, or received from, specified BGP neighbor addresses (destinations).

Syntax:
destinations
net address/net address net mask
advertised-to *network address*
received-from *network address*

Example: destination

```
Network/MaskLen    NextHop         MED   Weight LPref AAG AGRAS ORG AS-Path
142.4.0.0/16       192.0.251.165   100   0      0     No  0     IGP seq[165-178]
```

destinations *net address*

Displays detailed information on the specified route or destination network. The command shows how a specific route was learned, the best path to a specific destination, the metric associated with the route, and other information.

Example: destinations 142.4.0.0

```
Network/MaskLen   NextHop        MED  Weight LPref     AAG AGRAS ORG ASPath
142.4.0.0/16      192.0.251.165  100  0      0         No  0     IGP seq[165-178]

Dest:142.4.0.0/16, Age:180, Upd#:13,LastSent:0001:53:32

Eligible paths: 2
PathID: 8  (Best Path)
        ASpath: seq[165-178]
        Origin: IGP, Pref: 507, LocalPref: 0
        Metric: 0, Weight: 0, MED: 100
        NextHop: 192.0.251.165, Neighbor: 192.0.251.165
        AtomicAggr: No

PathID: 21
        ASpath: seq[168-165-178]
        Origin: IGP, Pref: 505, LocalPref: 0
        Metric: 0, Weight: 0, MED: 0
        NextHop: 128.185.250.168, Neighbor: 128.185.250.168
        AtomicAggr: No
```

ASpath
Enumeration of autonomous systems along the path.

-seq: Sequence of autonomous systems in order in the path

-set: Set of autonomous systems in the path.

Origin The originator of the destination. This is EGP, IGP, or Incomplete (originated by some other means not known).

LocalPref
The originating router's degree of preference for the destination.

Metric The path metric with which the route is imported.

Weight
The path weight.

MED A multi-exit discriminator value, used to discriminate among multiple entry/exit points to the same AS.

NextHop
The address of the router to use as the forwarding address for destinations reachable via the given path.

AtomicAggr
Indicates whether the router advertising the path has included the path in an atomic-aggregate.

destinations *net address net mask*
Displays detailed information on the specified route or destination network. The command shows how a specific route was learned, the best path to a specific destination, the metric associated with the route, and other information.

This command is useful in cases where multiple network addresses have the same prefix and different masks. In such cases, specifying the network mask narrows the scope of the information presented.

Example: `destinations 194.10.16.0 255.255.240.0`

```
Dest:194.10.16.0/21, Age:0, Upd#:3, LastSent:0002:00:00

Eligible paths: 1
PathID: 0 - (Best Path)
        ASpath:
        Origin: IGP, Pref: 0, LocalPref: 0
        Metric: 0, Weight: 0, MED: 0
        NextHop: 194.10.16.167, Neighbor: 194.10.16.167
        AtomicAggr: No, Aggregator AS167/194.10.16.167
```

destinations advertised-to *net address*
Lists all routes advertised to the specified BGP neighbor.

Example: `destinations advertised-to`

```
BGP neighbor address [0.0.0.0]? 192.0.251.165

        Destinations advertised to BGP neighbor 192.0.251.165

Network         NextHop         MED  Weight  LPref      AAG AGRAS ORG ASPath
194.10.16.0/20  194.10.16.167   0    0       0          No  167   IGP
192.0.190.0/24  192.0.251.165   0    0       0          No  0     IGP seq [165]
142.4.0.0/16    192.0.251.165   0    0       0          No  0     IGP seq [165-178]
143.116.0.0/16  128.185.250.168 0    0       0          No  0     IGP seq [168]
```

destinations received-from *net address*
Lists all routes received from the specified BGP neighbor.

Example: `destinations received-from`

```
BGP neighbor address [0.0.0.0]? 128.185.250.167

        Destinations obtained from BGP neighbor 128.185.250.167

Network             NextHop         MED Weight  LPref      AAG AGRAS ORG ASPath
194.10.16.0/20      128.185.250.167 0   0       0          No  167   IGP seq[167]
192.0.190.0/24      128.185.250.167 0   0       0          No  0     IGP seq[167-165]
142.4.0.0/16        128.185.250.167 0   0       0          No  0     IGP seq[167-165-178]
```

Dump Routing Tables

For a complete explanation of the **dump routing tables** command, refer to “Dump Routing Table” in the “Monitoring IP” chapter of *8371 Interface Configuration and Software User’s Guide*

Neighbors

Use the **neighbors** command to display information on all active BGP neighbors.

Syntax:

neighbors *internet address*

Example: **neighbors**

```
IP-Address       Status   State        DAY-HH:MM:SS  BGPID           AS     Upd#

128.185.252.168  ENABLD  Established  00000:48:52  128.185.142.168  168     16
192.0.190.178    ENABLD  Established  00002:01:49  142.4.140.178    178     16
192.0.251.167    DISBLD  Established  00002:01:45  194.10.16.167    167     16
```

IP-Address
Specifies the IP address of the BGP neighbor.

State Specifies the state of the connection. Possible states are:

Connect
Waiting for the TCP connection to the neighbor to be completed.

Active In the event of TCP connection failure, the state is changed to Active, and the attempt to acquire the neighbor continues.

OpenSent
In this state OPEN has been sent, and BGP waits for an OPEN message from the neighbor.

OpenConfirm
In this state a KEEPALIVE has been sent in response to neighbor’s OPEN, and waits for a KEEPALIVE/NOTIFICATION from the neighbor.

Established
A BGP connection has been successfully established, and can now start to exchange UPDATE messages.

BGP-ID
Specifies the neighbor’s BGP Identification number.

AS Specifies the neighbor’s AS number.

Upd# Specifies the sequence number of the last UPDATE message sent to the neighbor.

internet-address
Use the **neighbor** command to display detailed data on a particular BGP neighbor.

Example: **neighbor 192.0.251.167**

```
  Active Conn: Sprt:1026  Dprt:179    State: Established KeepAlive/Hold
  Time: 4/12
  Passve Conn: None
  TCP connection errors: 0          TCP state transitions: 0

  BGP Messages:     Sent       Received                    Sent
  Received
  Open:             1          1          Update:          11         11
```

```
Notification:     0         0         KeepAlive:        1828      1830
Total Messages:   1840      1842

Msg Header Errs:  Sent      Received                    Sent
Received
Conn sync err:    0         0         Bad msg length:   0         0
Bad msg type:     0         0

Open Msg Errs:    Sent      Received                    Sent
Received
Unsupp versions:  0         0         Unsupp auth code: 0         0
Bad peer AS ident:0         0         Auth failure:     0         0
Bad BGP ident:    0         0         Bad hold time:    0         0

Update Msg Errs:  Sent      Received                    Sent
Received
Bad attr list:    0         0         AS routing loop:  0         0
Bad wlkn attr:    0         0         Bad NEXT_HOP atr: 0         0
Mssng wlkn attr:  0         0         Optional atr err: 0         0
Attr flags err:   0         0         Bad netwrk field: 0         0
Attr length err:  0         0         Bad AS_PATH attr: 0         0
Bad ORIGIN attr:  0         0

Total Errors:     Sent      Received                    Sent
Received
Msg Header Errs:  0         0         Hold Timer Exprd: 0         0
Open Msg Errs:    0         0         FSM Errs:         0         0
Update Msg Errs:  0         0         Cease:            0         0
```

Parameter

Use the BGP **parameter** command to display installed BGP globals in the BGP system.

Syntax:

parameter

Example:

```
BGP> parameter

classless-bgp is enabled.
compare-med-from-diff-as is enabled.
IP-route-table-scan-timer value is 5 seconds.
```

Paths

Use the BGP **paths** command to display the paths stored in the path description data base.

Syntax:

paths

Example:

```
paths
PathId   NextHop    MED  AAG   AGRAS  RefCnt   ORG   ASPath
0        10.2.0.3    0   No     0      2       IGP
4        192.2.0.2   0   No     0      2       IGP    seq[2]
5        192.2.0.2   0   No     2      1       IGP    seq[2]
6        192.2.0.2   0   No     0      1       IGP    seq[2-1]
7        10.2.0.168  0   No     0      4       IGP
8        192.3.0.1   0   No     0      2       IGP    seq[1]
9        192.2.0.2   0   No     2      1       IGP    seq[2]
10       10.2.0.3    0   No     0      1       IGP
```

PathId

Path identifier

NextHop
The address of the router to use as the forwarding address for the destinations that can be reached via the given path.

MED The multi-exit discriminator used to discriminate among multiple entry/exit points to the same AS.

AAG Indicates if the path has been atomic-aggregated that is the router that is advertising the given path has selected less specific route over the more specific one when presented with overlapping routes.

AGRAS
Indicates the AS number of the BGP speaker that aggregated the routes.

RefCnt
Indicates the number of path entities referring to the descriptor.

ORG Specifies the originator of the advertised destinations in the given path: either EGP, IGP, or Incomplete (originated by some other means not known).

AS Path
Enumeration of autonomous systems along the path.

seq: Sequence of autonomous systems in order in the path.

set: Set of autonomous systems in the path.

Ping

For a complete explanation of the **ping** command, see “Ping” on page 362

Policy-List

Use the **policy-list** command to display the current installed policy for specific neighbor and usage statistics of each policy.

Example: policy-list

```
Neighbor address[0.0.0.0]? 192.0.251.167
Policy Type(Receive/Send/Origin)[All]?Receive
```

Display for neighbor based policy configuration:

```
Receive policy list for neighbor '192.0.251.167':
Idx T Prefix            Match OrgAS AnyAS MED   Weight LPref IGPmet Usage
1   I 0.0.0.0/0         Range 0     0     0     0      0     1      1
```

Display for AS based policy configuration:

```
Receive policy :
Idx Type  Prefix            Match OrgAS AdjAS IGPmetric Usage
1   INCL  0.0.0.0/0         Range 0     0     1         1
```

Example: policy-list

```
Neighbor address[0.0.0.0]? 192.0.251.167
Policy Type(Receive/Send/Origin)[All]?Send
```

Display for neighbor based policy configuration:

```
send policy list for neighbor '0.0.0.0': 192.0.251.167
Idx T Prefix            Match OrgAS AnyAS TAG   MED   ASpad Usage
1   I 0.0.0.0/0         Range 0     0     0     0     0     1
```

Display for AS based policy configuration

```
send policy :
Idx Type  Prefix            Match OrgAS AdjAS TAG   Usage
1   INCL  0.0.0.0/0         Range 0     0     0     1
```

Example: policy-list

```
Neighbor address[0.0.0.0]? 192.0.251.167
Policy Type(Receive/Send/Origin)[All]?Origin

Origin policy list for neighbor '0.0.0.0':
Idx T Prefix             Match TAG   Usage
1   I 0.0.0.0/0          Range 0     1
```

Reset Neighbor

Use the **reset neighbor** command to reset the specified BGP neighbor, based on the neighbor configuration parameters stored in the configuration memory.

Syntax:

reset neighbor *ip address*

Example: **reset neighbor**

```
Neighbor address[0.0.0.0]? 128.185.250.167
```

Sizes

Use the BGP **sizes** command to display the number of entries stored in the various data bases.

Syntax:

sizes

Example:

```
sizes

# Paths:                                11
# Path descriptors:                      7
  Update sequence#:                     22
# Routing tbl entries (allocated):       6
# Current tbl entries (not imported):    0
# Current tbl entries (imported to IGP): 3
```

Paths Total number of eligible paths for all the routes in the BGP routing table.

Path descriptors
Total number of path descriptors in the database used to hold common path information.

Update sequence#
Indicates the current update sequence number.

Routing tbl entries (allocated)
Indicates the number of entries in BGP routing table.

Current tbl entries (not imported)
Indicates the number of BGP routes not imported into IGP.

Current tbl entries(imported to IGP)
Indicates the number of BGP routes imported into IGP.

Traceroute

For a complete explanation of the **traceroute** command, see “Traceroute” on page 366.

Chapter 39. Configuring and Monitoring TCP/IP Host Services

This chapter describes how to configure the TCP/IP Host Services (TCP/IP Host) protocol and how to use the TCP/IP Host configuration commands. The chapter includes the following sections:

- "Accessing the TCP/IP Host Configuration Environment"
- "Basic Configuration Procedures"
- "TCP/IP Host Configuration Commands" on page 526
- "Accessing the TCP/IP Host Monitoring Environment" on page 529
- "TCP/IP Host Monitoring Commands" on page 529

See "TCP/IP Host Services (Bridge-Only Management)" on page 255 if you want to know more about why you would use TCP/IP host services.

Accessing the TCP/IP Host Configuration Environment

To access the TCP/IP Host configuration environment, enter the following command at the Config> prompt:

```
Config> protocol hst
TCP/IP-Host Services user configuration
TCP/IP-Host Config>
```

Basic Configuration Procedures

The following sections describe the basic configuration procedures for enabling TCP/IP Host Services on your IBM 8371.

Setting the IP Address

To minimally configure TCP/IP Host services, assign the IBM 8371 an IP address by using the **set ip-host** command. This IP address is associated with the IBM 8371 as a whole, instead of being associated with a single interface.

Enabling TCP/IP Host Services

Use the **enable services** command to enable TCP/IP Host Services.

Adding a Default Gateway

The IBM 8371 uses its default gateway to communicate with hosts and gateways that are not on the bridged network to which the IBM 8371 is directly connected. The IBM 8371 can dynamically learn its default gateway using either ICMP Router Discovery (see the **enable router-discovery** command in this chapter) or RIP (see the **enable rip-listening** command in this chapter). You can also statically specify one or more default gateways by using the **add default gateway** command. The IBM 8371 uses only one default gateway at a time; any additional default gateways are used for backup.

To save the assigned IP address and default gateway information,

© Copyright IBM Corp. 1996, 1999

1. Exit from the `TCP/IP-Host config>` prompt to the `Config>` prompt.
2. Use the **write** command at the `Config>` prompt to write the current configuration to memory.
3. Enter **CTRL-P** to get to the OPCON prompt and use the **reload** OPCON command to load a new copy of the software.
4. After reloading the IBM 8371, return to the `TCP/IP-Host config>` prompt.

TCP/IP Host Configuration Commands

This section describes the TCP/IP Host configuration commands. The TCP/IP Host configuration commands allow you to specify network parameters for the TCP/IP Host bridge. Restart the device to activate the configuration commands.

Note: The TCP/IP host configuration commands are not effective immediately. They remain pending until you reload the device.

Enter the TCP/IP Host configuration commands at the `TCP/IP-Host config>` prompt. Table 71 shows the commands.

Table 71. TCP/IP Host Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Adds a default-gateway.
Delete	Deletes a default-gateway.
Disable	Disables TCP/IP Host Services, router-discovery processes, and RIP listening.
Enable	Enables TCP/IP Host Services, router-discovery processes, and RIP listening.
List	Lists the current TCP/IP Host configuration.
Set	Sets the IBM 8371’s IP address.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Response to TCP/IP Host Configuration Commands

The TCP/IP host configuration (Talk 6) commands are not effective immediately. They remain pending until you issue the **reload** command.

Add

Use the **add** command to add default gateways (that is, routers) to your configuration.

Default gateways are used when trying to send packets to IP destinations that are off the local subnet. The routing table is then built up through redirect processing. An attempt is made to detect routers that disappear.

Syntax:

add default-gateway *def-gateway-IP-address*

Example: **`add default-gateway`**

```
Default-Gateway address [0.0.0.0]?  123.45.67.89
```

Delete

Use the **delete** command to delete default gateways from your IBM 8371 configuration. Enter the IP address of the default gateway you want to remove after the **delete** command.

Syntax:

delete default-gateway *def-gateway-IP-address*

Example: **`delete default-gateway`**

```
Enter address to be deleted [0.0.0.0]?  123.45.67.89
```

Disable

Use the **disable** command to disable the following TCP/IP functions:
- TCP/IP Host Services
- Router-discovery processes
- RIP listening

Syntax:

disable rip-listening
router-discovery
services

rip-listening
Disables the building of routing table entries that have been gathered by listening to the RIP protocol. By default, RIP-listening is disabled.

Example: **`disable rip-listening`**

router-discovery
Disables the ability to learn default gateways by receiving ICMP Router Discovery messages. By default, router discovery is enabled.

Example: **`disable router-discovery`**

services
Disables the TCP/IP Host Services protocol entirely. If IP routing is not enabled, TCP/IP Host Services is enabled by default.

Example: **`disable services`**

Enable

Use the **enable** command to enable the following TCP/IP functions:
- TCP/IP Host Services
- Router discovery processes
- RIP listening

Syntax:

enable rip-listening
router-discovery
services

rip-listening
Enables the building of routing table entries that have been gathered by the bridge “listening” to the RIP protocol. RIP-listening is disabled by default.

Example: `enable rip-listening`

router-discovery
Enables the learning of default gateways through reception of ICMP Router Discovery messages. By default, router discovery is enabled.

Example: `enable router-discovery`

services
Enables the TCP/IP Host Services protocol. If IP routing is not enabled, TCP/IP Host Services is enabled by default.

Example: `enable services`

List

Use the **list** command to display information about the current TCP/IP Host configuration.

Syntax:

list

Example: `list`

```
TCP/IP-Host config>list

TCP/IP Host SERVICES    : enabled
IP-HOST Address         : 128.185.142.1
        Mask            : 255.255.255.0
DEFAULT-GATEWAY Address : 128.185.142.47
RIP-LISTENING           : disabled
ROUTER-DISCOVERY        : enabled

TCP/IP-Host config>
```

TCP/IP Host SERVICES	Displays whether TCP/IP Host SERVICES is enabled or disabled.
IP-HOST Address	Displays the current IP-HOST Address.
IP-HOST Mask	Displays the current IP-HOST Mask.
DEFAULT-GATEWAY Address	Displays the current DEFAULT-GATEWAY Address.
RIP-LISTENING	Displays whether RIP-LISTENING is enabled or disabled.
ROUTER DISCOVERY	Displays whether ROUTER DISCOVERY is enabled or disabled.

Set

Use the **set** command to set the IBM 8371's IP address. You must assign the IBM 8371 an IP address before enabling TCP/IP Host Services.

Note: If the IP address is not already configured, it is set (by default) using boot information. This process applies only if the IBM 8371 is a network host operating as an IP host.

Syntax:

set ip-host address *IP-host-address*

Example: `set ip 123.45.67.89`

```
Address mask [255.255.0.0]?
IP-Host Address set.
```

Monitoring TCP/IP Host Services

This section describes how to monitor the TCP/IP Host Services on the IBM 8371.

Accessing the TCP/IP Host Monitoring Environment

To access the TCP/IP Host monitoring environment, enter the following command at the + (GWCON) prompt:

```
+ protocol hst
TCP/IP-Host>
```

TCP/IP Host Monitoring Commands

This section describes the TCP/IP Host monitoring commands. These commands allow you to view parameters and enter information requests from the active terminal. Enter these commands at the TCP/IP-Host> prompt. Table 72 shows the commands.

Table 72. TCP/IP Host Monitoring Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Dump	Displays the current IP routing table. One line is printed for each destination.
Interface	Displays the IBM 8371’s IP address.
Ping	Continuously pings a given destination, printing a line for each response received.
Traceroute	Displays the hop-by-hop route to a given destination.
Routers	Displays the list of all IP routers known to the IBM 8371.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Dump

Use the **dump** command to display the current IP routing table. One line is printed for each destination. Many of the entries that are displayed are the result of ICMP redirects.

Syntax:

dump

Example:

```
TCP/IP Host> dump
Type   Dest net         Mask      Cost     Age       Next hop(s)

Stat   0.0.0.0          00000000  0        51        128.185.142.47
 Dir*  128.185.142.0    FFFFFF00  1        50        BDG/0

Default gateway in use.
Type Cost     Age       Next hop
Stat 0        51        128.185.142.47

Routing table size: 768 nets (52224 bytes), 2 nets known
                    0 nets hidden, 0 nets deleted, 0 nets inactive
                    0 routes used internally, 766 routes free
```

Type	Route type which indicates how the route was derived: RIP - the route was learned through the RIP protocol. Stat - a statically configured route. Dir - a directly connected network or subnet.
Dest net	Displays the IP address of the destination network/subnet.
Mask	Displays the IP address mask.
Cost	Displays the Route Cost.
Age	For RIP routes displays the time, in seconds, since the route was refreshed. For other types of routes displays the time, in seconds, since the route was installed.
Next Hop	Displays the IP address of the next device on the path toward the destination host. Also displayed is the interface type used by the sending device to forward the packet.
Default gateway	Displays the IP address of the default gateway along with the route type, cost, age, and next hop information associated with that entry.
Routing table size	Displays the current size (in networks and bytes) of the current table. Also identifies the number of networks (nets) known to the host.

Interface

Use the **interface** command to display the IBM 8371's IP address. When TCP/IP Host Services are running over the bridge, a single address is displayed on the terminal as Bridge/0.

Syntax:

interface

Example:

```
TCP/IP Host> interface
Interface     MTU    IP Address(es)   Mask(s)          Address-MTU
  BDG/0      1500    128.185.142.16   255.255.255.0    Unspecified
```

Interface	Displays the type of interface. For TCP/IP Host Services, this is always BDG/0, indicating the bridge.
IP Address	Displays the IP address of the TCP/IP Host Services interface.
Mask	Displays the IP address subnet mask.

Ping

Use the **ping** command to make the device send ICMP Echo Requests to a given destination once a second ("pinging") and watch for a response. This command can be used to isolate trouble in an internetwork environment.

This process is done continuously, incrementing the ICMP sequence number with each additional packet. Matching received ICMP Echo responses are reported with their sequence number and the round trip time. The granularity (time resolution) of the round trip time calculation is platform-specific, and usually is around 20 milliseconds.

To stop the pinging process, type any character at the terminal. At that time, a summary of packet loss, round trip time, and number of unreachable ICMP destinations will be displayed.

When a multicast address is given as destination, there may be multiple responses printed for each packet sent, one for each group member. Each returned response is displayed with the source address of the responder.

The size of the ping (number of data bytes in the ICMP message, excluding the ICMP header), TTL value, and rate of pinging are all user-configurable. The default values are a size of 56 bytes, a TTL of 64, and a rate of 1 ping per second.

Syntax:

ping *destination source size ttl rate*

Example:

```
TCP/IP Host> ping
Destination IP address [0.0.0.0]? 128.185.142.11
Source IP address [128.185.142.16]?
Ping data size in bytes [56]?
Ping TTL [64]?
Ping rate in seconds [1]?
PING 128.185.142.16 -> 128.185.142.11: 56 data bytes, ttl=64, ... every 1 sec.
56 data bytes from 128.185.142.11: icmp_seq=0. ttl=254. time=0. ms
56 data bytes from 128.185.142.11: icmp_seq=1. ttl=254. time=0. ms
56 data bytes from 128.185.142.11: icmp_seq=2. ttl=254. time=0. ms
56 data bytes from 128.185.142.11: icmp_seq=3. ttl=254. time=0. ms
56 data bytes from 128.185.142.11: icmp_seq=4. ttl=254. time=0. ms
56 data bytes from 128.185.142.11: icmp_seq=5. ttl=254. time=0. ms

----128.185.142.11 PING Statistics----
6 packets transmitted, 6 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
```

Traceroute

Use the **traceroute** command to display the entire path to a given destination, hop by hop. For each successive hop, the traceroute command sends out three probes and prints the IP address of the responder along with the round trip time associated with the response. If a particular probe receives no response, an asterisk (*) is printed. Each line in the display relates to this set of three probes, with the left-most number indicating the distance from the device executing the command (in network device hops).

The traceroute is complete when the destination is reached, an ICMP Destination Unreachable message is received, or the path length reaches 32 network device hops.

Syntax:

traceroute *destination source size probes wait ttl*

Example:

```
TCP/IP Host> traceroute
Destination IP address [0.0.0.0]? 128.185.144.239
Source IP address [128.185.142.16]?
Data size in bytes [56]?
Number of probes per hop [3]?
Wait time between retries in seconds [3]?
Maximum TTL [32]?
TRACEROUTE 128.185.142.16 -> 128.185.144.239: 56 data bytes
 1 128.185.142.11 16 ms 0 ms 0 ms
 2 128.185.143.33 16 ms 0 ms 0 ms
 3 128.185.144.239 16 ms 0 ms 0 ms
```

In the display:

TRACEROUTE Displays the destination area address and the size of the packet being sent to that address.

1	The first trace showing the destination's NSAP and the round trip time it took the packet to reach the destination and return. The packet is traced three times.
Destination unreachable	Indicates that no route to the destination is available.
1 * * * 2 * * *	Indicates that the device is expecting some form of response from the destination, but the destination is not responding.

When a probe receives an unexpected result (see the previous output example), several indicators can be printed. These indicators are explained in the following table.

!N	Indicates that an ICMP Destination Unreachable (net unreachable) has been received.
!H	Indicates that an ICMP Destination Unreachable (host unreachable) has been received.
!P	Indicates that an ICMP Destination Unreachable (protocol unreachable) has been received.
!	Indicates that the destination has been reached, but the reply sent by the destination has been received with a TTL of 1. This usually indicates an error in the destination, prevalent in some versions of UNIX, whereby the destination is inserting the probe's TTL in its replies. This unfortunately leads to a number of lines consisting solely of asterisks before the destination is finally reached.

Routers

Use the **routers** command to display the list of all IP routers that are known to the IBM 8371. Routers can be learned through:

- Static configuration (using the **add default-gateway** command explained on page "Add" on page 526).
- Received ICMP redirects
- ICMP Router Discovery messages (if configured)
- RIP updates (if configured)

Each router is listed with its origin, its priority (used when selecting the default route), and its lifetime (the number of seconds before the router will be declared invalid unless it is heard from again).

Syntax:

routers

Example: `routers`

Chapter 40. Using SNMP

This chapter describes SNMP. It contains the following sections:

- "Network Management"
- "SNMP Management"

Network Management

Refer to the *8371 Server Introduction and Planning Guide* for information about Network Management.

SNMP Management

The IBM 8371 provides a Simple Network Management Protocol (SNMP) interface to network management platforms and applications, such as the Nways Campus Manager products.

SNMP is used for monitoring and managing IP hosts in an IP network and uses software called an SNMP agent to enable network hosts to read and modify some of the IBM 8371's operational parameters. In this way, SNMP establishes network management for the IP community.

You need to consider the following aspects of SNMP when you configure SNMP for your IBM 8371.

Community

The community allows you to define the IP address of the SNMP management station that is allowed to access the information in the SNMP agent's Management Information Base (MIB). You define a community name for use in accessing the MIB.

Authentication

The community name is used as an authentication scheme to prevent unauthorized users from learning information about an SNMP agent or modifying its characteristics.

This scheme involves defining one or more sets of MIB data (referred to as MIB views) and associating an access privilege (read-only, read-write), an IP mask, and a community name with each MIB view. The IP mask establishes which IP addresses can originate access requests for a given MIB view and the community name serves as a password that must be matched by the SNMP requests. The community name is included in each SNMP message and verified by the IBM 8371 SNMP agent. An SNMP request will be rejected if it does not provide the correct community name, does not match the IP mask, or attempts an access that is inconsistent with the assigned access privilege.

MIB Support

A MIB is a virtual information store that provides access to management information. This information is defined as MIB objects which can be accessed and, in some cases, be modified using network management tools.

© Copyright IBM Corp. 1996, 1999

IBM 8371 provides a comprehensive set of standard MIBS, enterprise-specific MIBs for monitoring and managing resources, and Readme files.

You can find the Readme files documenting IBM 8371 MIB support by accessing the appropriate release directory on the World Wide Web at URL:

- **ftp://ftp.nways.raleigh.ibm.com/pub/netmgmt/8371/**

To receive a copy of a specific MIB, enter the **get** command with the name of the MIB. For example, the command,**get rfc1213.txt** places a copy of the specified MIB in the directory from which you connected to the FTP server.

You can access the following information from the ftp site:
- Standard MIBs
- Enterprise MIBs
- SNMP generic traps
- Enterprise-specific MIBs
- Settable values

SNMP generic traps, Enterprise MIBS, and settable values are located in the Readme files.

All MIB objects are implemented as READ-ONLY objects even if their access clause is defined as read-write or read-create, except those MIB objects identified in the Readme file that support SETs for objects that have their access clause defined as read-write or read-create.

Trap Messages

Trap messages are unsolicited messages sent from the SNMP agent in the device to an SNMP manager in response to a device or network condition, such as a device reload or network down.

Chapter 41. Configuring and Monitoring SNMP

This chapter describes the SNMP configuring and monitoring commands. It includes the following sections:

- "Accessing the SNMP Configuration Environment"
- "SNMP Configuration Commands"
- "Accessing the SNMP Monitoring Environment" on page 544
- "SNMP Monitoring Commands" on page 544

Accessing the SNMP Configuration Environment

To access the SNMP configuration environment, enter the following command at the `Config>` prompt:

```
Config> protocol snmp
SNMP user configuration
SNMP Config>
```

SNMP Configuration Commands

This section describes the SNMP configuration commands.

Table 73 lists the SNMP configuration commands. The SNMP configuration commands allow you to specify parameters that define the relationship between the SNMP agent and the network management station. The information you specify takes effect immediately after a restart or reload of the IBM 8371.

Enter the SNMP configuration commands at the `SNMP Config>` prompt.

Table 73. SNMP Configuration Commands Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
Add	Adds a community to the list of SNMP communities, an IP address with mask to a community, or a subtree to a MIB view.
Delete	Removes a community from the list of SNMP communities, an IP address with mask from a community, or a subtree from a MIB view.
Disable	Disables SNMP protocol and traps associated with named communities.
Enable	Enables SNMP protocol and traps associated with named communities.
List	Displays the current communities with their associated access modes, enabled traps, IP addresses, and views. Also displays all views and their associated MIB subtrees.
Set	Sets a community's access mode or view. A community's access mode is one of the following: Read and trap generation Read, write and trap generation Trap generation only This command is also used to set a trap UDP port.

© Copyright IBM Corp. 1996, 1999

Table 73. SNMP Configuration Commands Summary (continued)

Command	Function
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **add** command to add a community name to the list of SNMP communities, add an address to a community, or assign a portion of the MIB (subtree) to a view.

Syntax:

add community
address
sub_tree

community

Use the **add community** command to create a community. It will be created with a default access of read_trap, a view of all, all traps disabled, and all IP addresses allowed.

Note: To select access type or trap control, use the **set community access** command to assign access types to existing SNMP communities and use the **enable trap** or the **disable trap** command for trap control.

community name

Provides the community name used by the SNMP client. This community name is used when accessing the management information base (MIB) in the device from the host specified by the Community IP address parameter.

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: none

Example:

```
SNMP Config>  add community
Community Name  []? comm01
Community added successfully
```

address

Use the **add address** command to add to the community definition an address of a network management station in the network that should be allowed to communicate with this box. You must supply the name of the community and the network address (in standard a.b.c.d notation). You also may supply a net mask to restrict access to either an individual host (mask = 255.255.255.255) or to a network of hosts. More than one address can be added to a community; enter the command each time you want to add another address.

If you do not specify an address for a community, requests are handled from any host.

Addresses also specify hosts that receive the traps. If no address is specified, no trap is generated.

community name

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: none

IP address

Valid Values: Any valid IP address.

Default Value: 0.0.0.0

ip mask

You also may supply a mask to restrict access to either an individual host (mask = 255.255.255.255) or to a network of hosts.

Valid Values: 0.0.0.0 - 255.255.255.255

Default Value: 255.255.255.255

Example:

```
SNMP Config> add address
Community Name []?
IP Address [0.0.0.0]?
IP Mask [255.255.255.255]?
```

sub_tree

Use the **add sub_tree** command to add a portion of the MIB to a view or to create a new view. The default is the entire MIB. The **add sub_tree** command is used to manage MIB views. More than one subtree can be added to a view defined by <view_text_name>.

view name

Specifies the name of the view to be created.

Valid Values: Any alphanumeric character string up to 31 characters in length. Characters such as spaces, tabs, or <Esc> key sequences are not accepted.

Default Value: none

Note: You must assign a view to one or more communities using the **set community view** command to have it take effect. The subtree definitions are inclusive; that is, the subtree OID specified and any OID that is lexicographically greater than the specified OID is considered part of the MIB view.

If a community is added using the **add community** command, all supported MIB views are assigned to the community unless the **set community view** command is used to assign specific views to the community.

MIB OID name

Specifies the MIB Object ID for the sub_tree. This must be entered as a numeric value, not a symbolic value.

This parameter contains a MIB subtree name included in the view defined with the View name parameter. All children of a specified MIB subtree are also included in the view.

For example, to provide a view that would give access to the system group in MIB-II, specify **1.3.6.1.2.1.1**.

Valid Values:

An object identifier in the form of <element1>.<element2>.<element3>. . ., where:

- You need a minimum of 1 element. Since all MIB OIDs begin with *1.3.6.1*, the minimum number of elements that you need to be provide in order for the view to differ from *all* is 5 (1.3.6.1.X).
- You can define a maximum of 31 characters, including the . separators.
- All elements after the first four (*1.3.6.1*) are integers between 0 and 127.

Note: This value must be numeric in dotted notation, *not* a symbolic value.

Default Value: none

Example:

```
SNMP Config> add sub_tree
View Name  []? view01
MIB OID name []? 1.3.6.1.1
Subtree added successfully
```

Delete

Use the **delete** command to delete a community and all of its addresses, a specific address, or a subtree from a view.

Syntax:

delete community

address

sub_tree

community

Removes a community and its IP addresses.

community name

Specifies a community name used by the SNMP client. This community name is used when accessing the management information base (MIB) in the device from the host specified by the Community IP address parameter.

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: none

Example:

```
SNMP Config> delete community
Community Name []?
```

address

Removes an address from a community. You must supply the name.

community name

Specifies the name of the community from which an address is to be removed. This community name is used when accessing the management information base (MIB) in the device from the host specified by the Community IP address parameter.

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: public

IP address

Specifies the IP address to be removed.

Valid Values: Any valid IP address.

Default Value: none

Example:

```
SNMP Config>  delete address
Community Name []?
IP address []?
```

sub_tree

Removes a MIB or a portion of the MIB from a view. You must supply the name of the subtree. If all subtrees are deleted, the MIB view is also deleted and all references to it from any associated SNMP communities are removed.

view name

Specifies the view used by the community defined in the **community name** parameter. This view determines which MIB objects this community may access. If no view is specified, the community may access all objects known to the device's SNMP agent.

This parameter should be answered if you decide to restrict a community from accessing the entire MIB managed by the device's SNMP agent.

Default Value: none

MIB OID name

Specifies the MIB Object ID for the sub_tree. This must be entered as a numeric value, not a symbolic value.

This parameter contains a MIB subtree name included in the view defined with the View name parameter. All children of a specified MIB subtree are also included in the view.

Valid Values: An object identifier in the form of <element1>.<element2>.<element3>. . ., where:

- You need a minimum of 1 element. Since all MIB OIDs begin with *1.3.6.1*, the minimum number of elements that you need to be provide in order for the view to differ from *all* is 5 (1.3.6.1.X).
- You can define a maximum of 31 characters, including the . separators.
- All elements after the first four (*1.3.6.1*) are integers between 0 and 127.

Default Value:nne

Example:

```
SNMP Config> delete sub_tree
View name[]?
MIB OID[]?
```

Disable

Use the **disable** command to disable the SNMP protocol or specified traps on the device.

Syntax:

disable snmp

trap

snmp Disables SNMP.

Example: `disable snmp`

trap *trap type*
Disables specified traps or all traps.

trap type
Specifies the type of trap to be disabled. Valid trap types are shown in Table 74.

community name

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: none

Example:

```
SNMP Config> disable trap link_up
Community name []?
```

Table 74. SNMP Trap Types

Trap Type	Description
all	Specifies all traps in a specified community.
cold_start	A cold start trap means that the transmitting device is reinitializing and that the agent's configuration or the protocol entity implementation may be altered.
warm_start	A warm start trap means that the transmitting device is reinitializing, but the configuration or protocol implementation will remain the same. Specify the community name as part of the command line.
link_down	A link_down trap recognizes a failure in one of the communication links represented in the agent's configuration. The link_down trap-PDU contains the name and value of the ifIndex instance for the affected link as the first element of its variable-bindings.
link_up	A link_up trap recognizes that a previously inactive link in the network has come up. The link_up trap-PDU contains the name and value of the ifIndex instance for the affected link as the first element of its variable-bindings.
auth_fail	Authentication failure traps indicate that the sender of the SNMP request does not have the proper permission to talk to this box's SNMP agent.
enterprise	Enterprise specific traps indicate that some enterprise specific event has occurred. The specific-trap field identifies the particular trap that occurred. For example, when configured to do so, ELS event messages are sent in enterprise-specific traps.

Enable

Use the **enable** command to enable the SNMP protocol or specified traps on the device.

Syntax:

enable snmp

trap

snmp Enables SNMP

Example: `enable snmp`

trap *trap type*
Enables specified traps or all traps.

trap type
Specifies the trap type to be enabled. Valid trap types are shown in Table 74 on page 540.

community name

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: none

List

Use the **list** command to display the current configuration of SNMP communities, access modes, traps, network addresses, and views.

Syntax:

list all

community

views

list all Displays the current configuration of SNMP communities for Access, Traps, Address, and View. See the description of the **list community** command for details on the options.

Example: `list all`

```
SNMP Config>list all

   SNMP is enabled
   Trap UDP port: 162
   SRAM write is enabled

            Community Name                Access
   -------------------------------  -------------------
   oxnard                           Read, Write, Trap
   public                           Read, Trap


           Community Name             IP Address         IP Mask
   -------------------------------  ---------------  ---------------
   oxnard                           1.1.1.2          255.255.255.255
   public                           All              N/A


           Community Name                Enabled Traps
   -------------------------------  -------------------------------
   oxnard                           Link Down, Cold Restart
   public                           None


            Community Name                    View
   -------------------------------  -------------------------------
   oxnard                           mib2
```

```
public                          All

          View Name                          Sub-Tree
------------------------------ ------------------------------
mib2                           1.3.6.1.2
```

list community *option*

Displays the current attributes of an SNMP community. Options are access, address, traps, view.

Option	Description
Access	Displays the access modes for the community.
Address	Displays the network address for the community.
Traps	Displays the types of traps generated for the community.
View	Displays the MIB view for the community.

Example:

```
SNMP Config  list community access

            Community Name   Access
            public           Read, Write, Trap
            oxnard           Read, Trap
```

Example:

```
SNMP Config>  list community address

            Community Name   IP Address   IP Mask
            public           All          N/A
            oxnard           1.1.1.2      255.255.255.255
```

Example:

```
SNMP Config  list community traps

            Community Name   Enabled Traps
            public           Link Down, Cold Restart
            oxnard           NONE
```

Example:

```
SNMP Config>  list community view

             Community Name   View
            public           All
            oxnard           mib2
```

list views

Displays the current views for a specified SNMP community.

Example:

```
SNMP Config  list views

            View Name   Sub-Tree
            mib2        1.3.6.1.2.1
```

Set

Use the **set** command to assign a MIB view to a community, to set the SNMP UDP trap port number, or set the access mode of the community.

Syntax:

set community access

community view

trap_port

community access

Use the **set community access** command to assign one of three access types to a community. You must supply the name of the community and the access type.

options

Choose an option from the following list:

read_trap

Allows read access and trap generation to the named community.

write_read_trap

Allows write and read access and trap generation to the community specified.

trap_only

Indicates the community is used only when sending an SNMP trap.

comm_name

The **community name** has:
Valid Values: A string of 1 to 31 alphanumeric characters.

Characters such as spaces, tabs, or <ESC> key sequences are not supported.
Default Value: none

Example: `set community access <options> comm_name`

community view

Use the **set community view** command to assign a MIB view to a community.

comm_name

Valid Values: A string of 1 to 31 alphanumeric characters. Characters such as spaces, tabs, or <ESC> key sequences are not supported.

Default Value: none

all Allows access to all MIB objects for the named community. All is the default.

view_text_name

Assigns a specified MIB view to the named community.

Example: `set community view comm_name <all` *or* `view_text_name>`

trap_port

Use the **set trap_port** command to specify a UDP port number, other than the default standard port 162, to send traps to.

Default Value: standard port

Example: `set trap_port udpport#`

UDP Port Number

Specifies a User Datagram Protocol port other than the standard UDP port.

Default Value: 162

Accessing the SNMP Monitoring Environment

To access the SNMP monitoring environment, enter the following command at the **+** (GWCON) prompt:

```
+ protocol snmp
SNMP>
```

SNMP Monitoring Commands

This section describes the SNMP monitoring commands.

Table 75 lists the SNMP monitoring commands. The SNMP monitoring commands allow you to view the parameters of the SNMP configuration and display some statistics relating to the SNMP agent.

Temporary changes to the runtime SNMP parameters can be made through the monitoring. If you want to make the temporary changes permanent, then use the SAVE command. If the original SNMP configuration needs to be restored, use the **revert** command. This command erases the specified changes and restores the settings to the values in the permanent SNMP configuration.

Enter the SNMP monitoring commands at the SNMP> prompt.

Table 75. SNMP Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Add	Adds a community to the list of SNMP communities, an IP address with mask to a community, or a subtree to a MIB view.
Delete	Removes a community from the list of SNMP communities, an IP address with mask from a community, or a subtree from a MIB view.
Disable	Disables traps associated with named communities. Disabling SNMP or SRAM_write must be done using the SNMP Config> configuration environment.
Enable	Enables traps associated with named communities. Enabling SNMP or SRAM_write must be done using the SNMP Config> configuration environment.
List	Displays the current configuration of SNMP communities, views, access modes, traps, and network addresses.
Reset	Updates the SNMP configuration with the values in the currently stored SNMP configuration.
Save	Takes the specified changes and saves then permanently in the SNMP configuration.
Set	Sets a community’s access mode or view. A community’s access mode is one of the following: • Read and trap generation • Read, write and trap generation • Trap generation only Also allows setting of trap UDP port.
Statistics	Displays statistics about the SNMP agent.
Revert	Erases dynamic changes and restores the settings to the values in the permanent SNMP configuration.
Reset	Updates the SNMP configuration with the values in the current stored SNMP configuration.

Table 75. SNMP Monitoring Command Summary (continued)

Command	Function
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Add

Use the **add** command to add a community name to the list of SNMP communities, add an address to a community, or assign a portion of the MIB (subtree) to a view.

For information on using the **add** command, see “Add” on page 536.

Delete

Use the **delete** command to delete:
- A specific address.
- A community and all of its addresses.
- A subtree from a view.

For information on using the **delete** command, see “Delete” on page 538.

Disable

Use the **disable** command to disable specified traps on the device.

For information on using the **disable** command, see “Disable” on page 540.

Enable

Use the **enable** command to enable specified traps on the device.

For information on using the **enable** command, see “Enable” on page 540.

List

Use the **list** command to display the current configuration of SNMP communities, views, access modes, traps, and network addresses.

Syntax:

list	all
	community
	views

For information about using the **list** command, see “List” on page 541.

Reset

Use the SNMP **reset** command to update the SNMP configuration with the values in the current stored SNMP configuration. This action allows changes to the current SNMP configuration when the device is restarted or reloaded.

Save

Use the **save** command to permanently save the specified changes.

Set

For information on using the **set** command, see “Set” on page 542.

Statistics

Use the **statistics** command to display statistics about the SNMP agent.

Syntax:

statistics

Example: `statistics`

```
              Max       Current    Current
             Alloc      Alloc       In Use

SNMP agent:  512000     181144     133120

SNMP MIBs:   1048576     57976      19712
```

The following information is displayed:

Max Alloc
The maximum amount of memory (in bytes) that is reserved for the SNMP component.

Current Alloc
As memory is needed, it is taken from the reserved pool (designated by MAX ALLOC) and moved in to an ″active″ memory pool. The size of this ″active″ memory pool size is indicated by the CURRENT ALLOC value.

Current In Use
This value represents the memory currently allocated from the ″active″ memory pool (designated by CURRENT ALLOC) that is in use by the SNMP component.

Chapter 42. Using MultiProtocol Over ATM (MPOA)

This chapter describes how to use Multiprotocol over ATM (MPOA) and includes the following section:

- "MPOA Overview"

MPOA Overview

The concept of virtual router, as shown in Figure 34, allows you to implement a conventional edge router function using MPOA servers, MPOA clients, and an ATM backbone network.

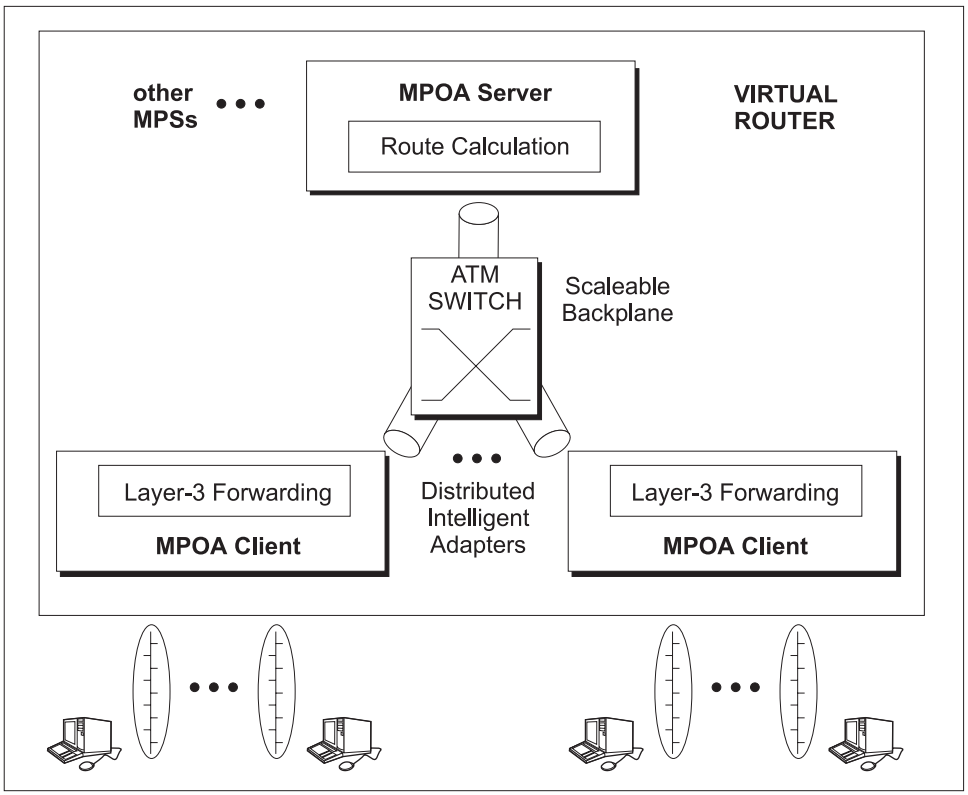

Figure 34. MPOA Virtual Router

MPOA uses networking technologies, such as bridging, LAN Emulation, and Next Hop Resolution Protocol, to implement the virtual router concept. As shown in Figure 35 on page 548, the virtual router model has:

- One router to manage
- One device participating in routing topology protocols, leading to simple edge devices
- Forwarding capacity of multiple devices

while a conventional edge router model has:

- Multiple routers to manage
- Multiple devices participating in routing topology protocols, leading to complex edge devices

© Copyright IBM Corp. 1996, 1999

- Forwarding capacity of multiple devices

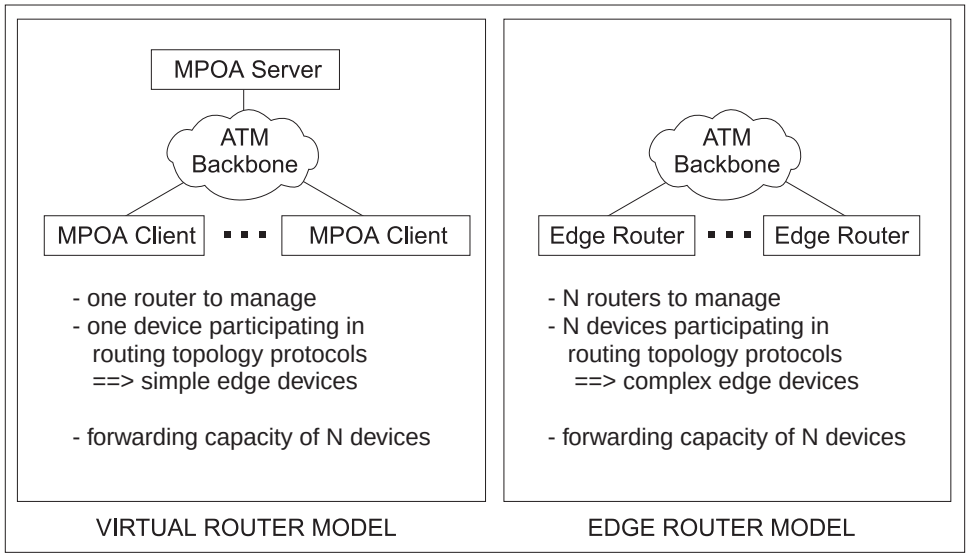

Figure 35. Comparison of Virtual Router and Edge Router Models

MPOA implements a virtual router with client/server protocols. MPOA clients (MPCs) issue requests to MPOA servers (MPSs). MPSs perform route calculations, while MPCs act as distributed intelligent adapters performing high-speed forwarding and the ATM network provides backplane throughput. MPSs are located with router functions and a NHRP server, while MPCs reside in MPOA hosts or MPOA edge devices, as shown in Figure 36 on page 549. The functions performed by a MPC in a MPOA host are very similar to those performed by a MPC in a MPOA edge device: establishing shortcut VCCs and forwarding intersubnet traffic over these VCCs to improve system performance. All MPOA devices include a LAN Emulation Client (LEC) that provides default path interconnection.

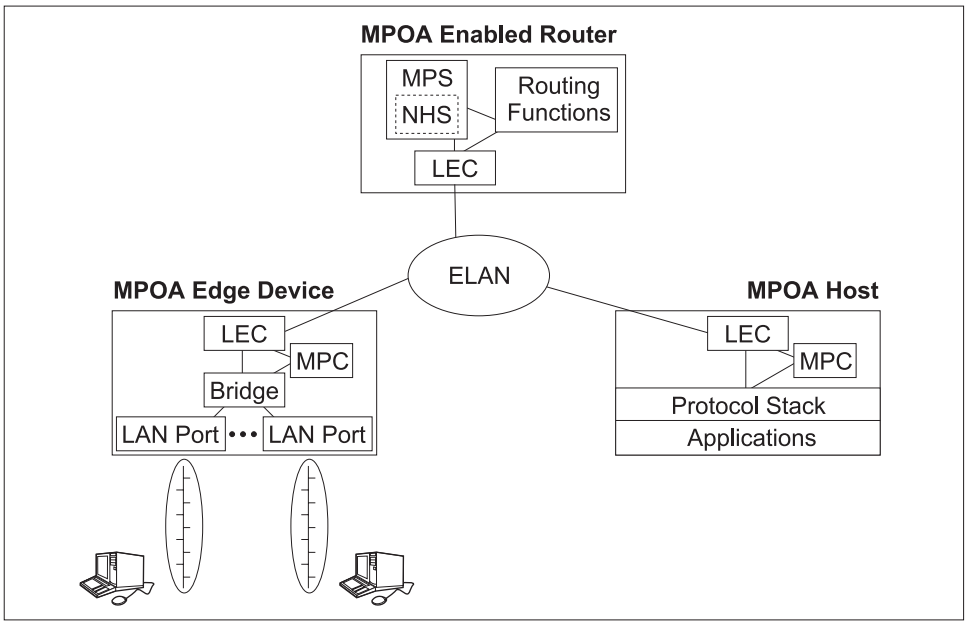

Figure 36. MPOA Components

MPOA and LAN Emulation

MPOA relies on LAN Emulation for three important functions:

- Auto-configuration
- Dynamic Device Discovery
- Intersubnet/default path connectivity

Auto-configuration allows MPOA configuration parameters to be stored and distributed from the LAN Emulation Configuration Server, or LECS. MPOA devices can obtain their configuration information from the LECS while they are being initialized, so that individual device configuration is reduced. See the chapter entitled “Configuring and Monitoring LAN Emulation Services” in the *8371 Interface Configuration and Software User’s Guide* for additional information about configuring LECS.

MPOA devices dynamically learn about neighbor components through the discovery protocol. MPOA devices attach special TLVs to LAN Emulation control messages and then inspect received TLVs to identify MAC addresses associated with other MPOA devices. Refer to the chapter entitled “Overview of LAN Emulation” in the *8371 Interface Configuration and Software User’s Guide* for additional information about LAN Emulation TLVs.

MPOA clients bridge intrasubnet traffic over ELANs. Since most MPOA edge devices include LAN switching hardware capabilities, intrasubnet traffic is handled with end-to-end switching. This use of bridging, coupled with dynamic device discovery, enables the MPC to be independent of router topology while maintaining the change management benefits provided by VLANs. For example, a station can be moved from a segment behind one MPC to a segment behind another MPC without any reconfiguration.

MPOA and Shortcut Establishment

MPOA clients are responsible for initiating shortcut establishment. The MPC discovers the MAC addresses of the MPS routers and the corresponding ATM addresses. MPC then monitors traffic flow to these MAC addresses, and when the flow exceeds a configured threshold, MPC initiates shortcut establishment by sending a MPOA resolution request to the associated MPS.

The MPOA implementation supports shortcuts for IP and IPX traffic.

Chapter 43. Configuring and Monitoring MPOA

This chapter describes how to use the MPOA configuration and operating commands and includes the following sections:

- "Accessing the MPOA Configuration Environment"
- "MPC Configuration Commands"
- "Accessing the MPOA Monitoring Environment" on page 557
- "MPC Monitoring Commands" on page 558

Accessing the MPOA Configuration Environment

Use the following procedure to access the MPOA configuration process.

1. At the OPCON prompt, enter **talk 6**. (For more detail on this command, refer to *The OPCON Process and Commands* in the 8371 Interface Configuration and Software User's Guide.)

   After you enter the **talk 6** command, the CONFIG prompt (`Config>`) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.
2. At the CONFIG prompt, enter the **protocol mpoa** command to get to the `MPOA Config>` prompt.

MPOA Configuration Commands

The MPOA main menu includes the following commands.

Table 76. MPOA Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
MPC	Enters the MPC configuration environment. for the MPC instance defined over a specified ATM device. See "MPC Configuration Commands" for additional information.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

MPC Configuration Commands

To access the MPC *configuration* process, enter **mpc** *device#* at the `MPOA Config>` prompt to access the `MPC Config>` prompt. If you do not enter the *device#*, you will be prompted to supply the ATM device number.

Enter the following commands at the `MPC Config>` prompt. These commands apply to the MPC instance defined over the ATM device number supplied when you entered the `MPOA Config>` **mpc** *device#* command.

Table 77. MPC Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.

© Copyright IBM Corp. 1996, 1999

Table 77. MPC Configuration Command Summary (continued)

Command	Function
Add	Adds an MPC instance with default parameter values.
List	Lists the enabled/disabled status of the MPC instance.
Config	Allows explicit configuration of MPC parameters.
Remove	Removes a MPC configuration.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Note: The MPOA client does not have to be explicitly configured in order to start functioning at startup. If no explicit configuration of the MPC has been done under `talk 6`, an MPOA client is automatically created in the *enabled* state with a default set of parameter values and begins MPOA operation including shortcut initiation. You should explicitly configure any non-default configuration parameters. To prevent the MPOA client function from automatically being activated, you should use the **config** command to access the `MPC Configuration>` prompt and then use the **disable** command to create an MPC instance with a status of *disabled*.

Add

Use the **add** command to add a MPC instance with default parameters.

The **add** option requires that an ATM interface has been previously added.

The added MPC defaults to ***enabled***.

Note: When an MPC is created, it is automatically associated with all LECs on the ATM device that have a bridge port configured on them. There is no explicit configuration to associate particular LECs to the MPC. Further, this association is formed during startup time and not during configuration. Thus, even if no bridge ports have been defined at the time the MPC is added and configured, the MPC will still be associated with all LECs that have a bridge port associated with them at startup time. You cannot dynamically disable association of the MPC with a particular LEC at runtime.

Syntax:
add MPC

Remove

Use the **remove** command to remove a MPC configuration.

Syntax:
remove MPC

List

Use the **list** command to display the existing MPC instance.

Syntax:
list

Config

Use the **config** command to access the `MPC Configuration>` prompt and perform explicit configuration of the MPC parameters.

Syntax:
config

To configure MPC parameters explicitly, enter the following commands at the `MPC Configuration>` prompt.

Table 78. MPC Explicit Configuration Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
Enable	Enables the MPC instance.
Disable	Disables the MPC instance. This command can also be used to create an MPC instance with a status of ***disabled***.
Set	Sets explicit values for MPC configuration parameters.
List	Displays all the configuration information associated with the MPC instance.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Enable

Use the **enable** command to enable an MPC instance.

Syntax:
enable

Disable

Use the **disable** command to disable an MPC instance.

This command can also be used to create an MPC instance with a status of ***disabled***.

Note: An MPC instance will be automatically created at startup, even if no MPC has been explicitly added. Use the **disable** command to disable this MPC instance if you do not want to configure MPC.

Syntax:
disable

Set

Use the **set** command to explicitly configure MPC parameters.

Syntax:
set frame-count
frame-time
initial-retry-time
maximum-retry-time
hold-down-time
vcc-timeout
accept-config-from-lecs
fragmentation-mode

esi
selector
pcr
max-reserved-bandwidth
shortcuts
ip-protocol
ipx-protocol

frame-count

Controls the rate of traffic required before the MPC will initiate a shortcut to a given protocol destination. The MPC will initiate a shortcut when at least this many frames are forwarded to the same protocol destination for a period of ***frame-time*** seconds.

Valid Values: 1 to 65535 frames

Default Value: 10

frame-time

Controls the rate of traffic required before the MPC will initiate a shortcut to a given protocol destination. The MPC will initiate a shortcut when at least ***frame-count*** frames are forwarded to the same protocol destination for a period of this number of seconds.

Valid Values: 1 to 60 seconds

Default Value: 1

initial-retry-time

Specifies the initial value of the retry timer when the MPC sends a request. If a corresponding reply is not received before the expiration of the retry timer, the request is retried if the retry timer is less than ***maximum-retry-time*** and the retry timer is then doubled. This process will repeat until a reply is received or until the retry timer ≥ ***maximum-retry-time***.

Valid Values: 1 to 300 seconds

Default Value: 5

maximum-retry-time

Specifies the maximum value of the retry timer when the MPC sends a request. If a corresponding reply is not received before the expiration of the retry timer, the request is retried if the retry timer is less than this value and the retry timer is then doubled. This process will repeat until a reply is received or until the retry timer ≥ this value.

Valid Values: 10 to 300 seconds

Default Value: 40

hold-down-time

Specifies the minimum time to wait before re-initiating a failed resolution request.

Valid Values: 30 to 1200 seconds

Default Value: 160

vcc-timeout

Specifies the length of time after which either inactive control or inactive data connections will be cleared.

Valid Values: 1 to 10080 minutes

Default Value: 20

accept-config-from-lecs
Specifies whether configuration parameters received from the LECS will be accepted by the MPC.

Valid Values: yes or no

Default Value: yes

fragmentation-mode
Controls the manner that the ingress MPC handles IP packet fragmentation.

When this parameter is set to ***maximize-shortcut-usage***, frames requiring fragmentation will be sent to the MPOA server, while smaller frames will be sent over the shortcut. A potential consequence of using ***maximize-shortcut-usage*** is that packets can get out of order.

When this parameter is set to ***maximize-inorder-usage***, usage of a particular shortcut will be suspended for the ***hold-down-time*** if a frame requiring fragmentation is received, causing all frames for the destination to be sent to the MPOA server.

When this parameter is set to ***perform-fragmentation***, IP frames requiring fragmentation are fragmented by the MPC and then sent over the shortcut. Both shortcut usage and inorder delivery are maximized.

Note: A single flow requiring fragmentation can impact the performance of all flows.

Valid Values: maximize-shortcut-usage, maximize-inorder-packet-delivery, or perform-fragmentation

Default Value: perform-fragmentation

esi Specifies the ESI that is to be used as the ESI component of the MPC's ATM address. The MPC implementation uses a single ATM address for control as well as data VCCs, so this ATM address refers to both the control and data ATM addresses of the MPC.

Valid Values:

- Burned-in
- One of the set of enabled ESI definitions for the MPC ATM device

Default Value: Burned-in

selector
Specifies the selector value that is to be used in combination with the esi to create a value that is unique among all protocol components using the MPC ATM device.

Valid Values: any single valid octet value that has not already been used

Default Value: automatically created

pcr Specifies the desired peak cell rate for connections established by the MPC over the associated ATM device.

All connections established by the MPC are best-effort connections.

Valid Values: 0 - line speed of the ATM device (integer Kbps)

Default Value: line speed of the ATM device

max-reserved-bandwidth
Specifies the maximum amount of reserved bandwidth acceptable on incoming calls received over the associated ATM device.

Valid Values: 0 - line speed of the ATM device (integer Kbps)

Default Value: 0

shortcuts
Specifies whether the MPC should establish shortcuts to LANE devices over the associated ATM device.

Valid Values: yes or no

Default Value: yes

If the value of this parameter is *yes*, you will be prompted for the following additional information:

Choice of source address for LANE shortcuts
Specifies what source MAC address is to be used in frames transmitted on LANE shortcut VCCs.

Valid Values:
- The MAC address burned into the MPC's ATM device
- A locally-administered MAC address
- The MAC address provided in the MPOA resolution reply

Default Value: Burned-in MAC address

If you choose to provide a locally-administered MAC address, you will be prompted for the value to be used.

Valid Values: 12 hexadecimal digits in the range of X'400000000000' and X'7FFFFFFFFFFF'

Default Value: None

ip-protocol
Permits enabling or disabling of the MPOA protocol for IP traffic.

Valid Values: Yes or No

Default Value: Yes

ipx-protocol
Permits enabling or disabling of the MPOA protocol for IPX traffic.

Valid Values: Yes or No

Default Value: Yes

List

Use the **list** command to display configuration information about the existing MPC instance.

Syntax:
list

```
MPC Configuration> list
   MPC Configuration
   -----------------
   STATUS:                              ENABLED
   Shortcut Setup Frame Count:           10 (sec)
```

```
Shortcut Setup Frame Time:          1  frame (sec)
Initial Retry Time:                 5  (sec)
Maximum Retry Time:                 40 (sec)
Hold Down Time:                     160(sec)
VCC Timeout Period:                 20 (min)
Accept Config Parms from LECS       YES
Fragmentation Mode                  Maximize Shortcut Usage

Interface:                          36
ESI:                                Burned In ESI
Selector:                           3
Desired PCR:                        155000 (Kbps)
Maximum Reserved Bandwidth:         10000 (Kbps)
Line Rate:                          155 Mbps
Enable LANE Shortcuts:              Yes
Source MAC Address for Shortcuts:   Burned In
IP-Protocol:                        Enabled
IPX-Protocol:                       Disabled
```

Accessing the MPOA Monitoring Environment

Use the following procedure to access the MPOA monitoring commands. This gives you access to the MPOA *monitoring* process.

1. At the OPCON prompt, enter **talk 5**. (For more detail on this command, refer to *The OPCON Process and Commands* in the 8371 Interface Configuration and Software User's Guide.)

   After you enter the **talk 5** command, the GWCON prompt (+) displays on the terminal. If the prompt does not appear when you first enter configuration, press **Return** again.
2. At the + prompt, enter the **protocol MPOA** command to get you to the `MPOA>` prompt.

MPOA Monitoring Commands

The MPOA main menu includes the following commands.

Table 79. MPOA Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See "Getting Help" on page 10.
MPC	Enters the MPC monitoring environment of the MPC instance defined on the specified ATM device. See "MPC Monitoring Commands" on page 558 for additional information.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

MPC Monitoring Commands

To access the MPC *monitoring* process, enter **mpc** *device#* at the MPOA> prompt to access the MPC Console> prompt. Enter these commands at the MPC Console> prompt.

Table 80. MPC Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
atm-interface	Accesses the MPC-ATM> command prompt from which information about the ATM interface can be displayed.
mpc-base	Accesses the MPC-BASE> command prompt from which information about the overall MPC status can be displayed.
neighbor-mps	Accesses the MPC-MPS> command prompt from which information about the MPOA servers (MPS) that have been discovered by the MPC can be displayed.
VCCs	Accesses the MPC VCC> command prompt from which information about the VCCs being used by the MPC can be displayed.
ingress-cache	Accesses the MPC Ingress> command prompt from which information about the MPC’s ingress cache can be displayed.
egress-cache	Accesses the MPC egress> command prompt from which information about the MPC’s egress cache can be displayed.
configure	Accesses the MPC Configure> command prompt from which MPC configuration parameters can be dynamically changed.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Monitoring Commands for the MPC ATM-Interface

Enter the following commands at the MPC-ATM> command prompt.

Table 81. MPC ATM-Interface Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
display-interface-state	Provides information about the state of the MPC’s ATM interface and ATM address registration.
interface-statistics	Displays statistics about the ATM interface.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

Display-Interface-State

Use the **display-interface-state** command to provide information about the state of the MPC’s ATM interface and ATM address registration.

Syntax:
display-interface-state

Example:

```
MPC ATM>display
  MPOA Client Configured on ATM Interface  36:
  ============================================
```

```
1. ATM Interface Up/Down ?:              UP
2. ATM Address Activated By Switch ?:    TRUE
3. LLC Call Sap Ready ?:                 TRUE
4. LANE Call Sap Ready ?:                TRUE
5. Local ATM address :
     39.84.0F.00.00.00.00.00.00.00.00.00.02.10.00.5A.01.9C.00.03
```

Interface-Statistics

Use the **interface-statistics** command to display statistics such as the total number of address activation attempts and the number of times the ATM interface has been down.

Syntax:
interface-statistics

Example:

```
MPC ATM>inter
ATM Interface Statistics For This MPC:
-------------------------------------
Total Address Registration Timeouts:    0
Total Address Registration Failures:    0
Total Address Deactivations :           0
Total Net Downs:                        0
```

MPC Base Monitoring Commands

Enter the following commands at the MPC-BASE> command prompt.

Table 82. MPC BASE Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
list-config	Displays the current MPC configuration parameters.
LECs	Displays a list of local LAN Emulation Clients that are currently associated with the MPC.
state	Displays the current state of the MPC and the time since the last state change.
mpc-statistics	Displays a set of statistics for the MPC as a whole.
Enable-mpc	Dynamically starts a disabled MPC instance.
Enable-protocol	Dynamically starts a disabled MPC instance over IP or IPX.
Disable-protocol	Dynamically deactivates the MPC instance over IP or IPX.
Disable-mpc	Dynamically deactivates the MPC instance.
Create-mpc	Dynamically creates a new MPC instance.
Delete-mpc	Deletes the MPC instance.
clear-statistics	Resets all the statistics maintained for the MPC instance to their initial values.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

List-config

Use the **list-config** command to display the current configuration parameters of the MPC.

Syntax:
list-config

Example:

Note: This command displays the parameters that are configurable through `talk 6` as well as some parameters, such as the *ATM address packet trace filter* value that can only be configured through `talk 5` (in the CONFIGURE menu).

```
MPC Base>list
   MPOA Client Configuration:
========================

Status:                           ENABLED
Shortcut Setup Frame Count:       1
Shortcut Setup Frame Time:        1      (sec)
Initial Retry Time:               5      (sec)
Maximum Retry Time:               40     (sec)
Hold Down Time:                   160    (sec)
VCC Timeout Period:               20     (min)
Accept Config From LECS:          Yes
Fragmentation Mode:               Maximize Shortcut Usage
Interface:                        0
ESI:                              Burned In ESI
Selector:                         3
Desired PCR:                      155000   (kbps)
Maximum Reserved Bandwidth:       155000   (kbps)
Line Rate:                        155      (Mbps)
Enable LANE Shortcuts:            TRUE
Source MAC Address for Shortcuts: Burned In

Packet Trace Filtering Parameters:
================================

ATM Address Pkt Trace Filter Value
 00000000000000000000000000000000000000000000
 ATM Address Pkt Trace Filter Mask
 00000000000000000000000000000000000000000000
LAN Pkt Trace Filter Value
 000000000000
LAN Pkt Trace Filter Mask
 000000000000
```

LECs

Use the **LECs** command to display a list of local LAN Emulation Clients (LECs) that are currently associated with the MPC. For each LEC, the interface number, ELAN ID, ELAN type, and bridge port type are displayed.

Syntax:
lecs

Example:

```
LECs Associated w/  MPOA Client (interface 36):
================================================

     1) LEC Interface Number: 40
ELAN Type: ETHRNET      ELAN ID: x0
Bridge Port Type: TB PORT

Lan Destinations Registered by this LEC:
----------------------------------------
```

State

Use the **state** command to display the current state of the MPC and the time since the last state change.

Syntax:
state

Example:

```
MPC Base>state
MPOA Client State:
==================

ATM Interface Number:                  36
State:                                 MPC UP STATE
Time Since Last State Change (h:m:s):  00:33:40
Last (internal) error code:            0
  Network-layer Protocols enabled:       IP IPX
```

MPC-Statistics

Use the **mpc-statistics** command to display aggregate statistics for the MPC instance.

Syntax:
mpc-statistics

Example:

Note: This command is basically a combination of the **statistics** commands in each of the other submenus.

```
MPC Base>mpc
MPOA Client Statistics (interface 36):
======================================
  Ingress MPC Statistics For This MPC:
  ------------------------------------
Total Resolution Requests Sent:         7
Total Refresh Res. Requests Sent:       6
Total Res. Rqst Retransmissions:        1
Total Res. Rqst Timeouts:               0
Total Res. Reply Successes:             7
Total Res. Reply NAKs:                  0
Total Res. Replies Discarded:           0
Total MPS Purges Recvd:                 0
Total MPS Purged Mappings:              0
Total MPS Purges Discarded:             0
Total Triggers Recvd:                   0
Total Triggers Discarded:               0
Total Keep Alives Recvd:                218
Total Inactive Mappings Deleted:        0
Total Frames Forwarded On Shortcuts:    2174
Total Data Plane Purges Recvd:          0
Total Data Plane Purged Mappings:       0
Total Data Plane Purges Discarded:      0
Total NHRP Purge Replies Transmitted:   0



  Egress MPC Statistics For This MPC:
  -----------------------------------
Total Imposition Requests Recvd:        8
```

```
Total Imposition Rqsts NAKed:           0
Total Imposition Updates Received:      7
Total Imposition Purges Received:       0
Total Imposition Purged Mappings:       0
Total E-MPC Purge Rqsts Sent To MPSs:   0
Total E-MPC Purge Rqst Retransmissions: 0
Total E-MPC Purge Rqst Timeouts:        0
Tot. Frames Recvd & Fwded (Software):   2286
Total Recvd Frames Discarded:           0
Total Data Plane Purge Rqsts Sent:      0
Total Data Plane Purge Rqst Retransmits:0
Total Data Plane Purge Rqst Timeouts:   0
Total Egress Cache Entries Aged Out:    0


  VCC Statistics For This MPC:
  ----------------------------
Total Call Setup Failures:              0
Total Incoming Calls Rejected:          0
Total Connections Released Locally:     0
Total Calls Placed Successfully:        1
Total Calls Received Successfully:      1
Total Remote Hangups (Normal):          0
Total Remote Hangups (Error):           0


  ATM Interface Statistics For This MPC:
  --------------------------------------
Total Address Registration Timeouts:    0
Total Address Registration Failures:    0
Total Address Deactivations :           0
Total Net Downs:                        0


  Additional Misc. Stats
  ----------------------
Total Error Indication Frames Received: 0
Total Error Indication Frames Txmtd:    0
Total Invalid Frames Received:          0
Total Keep-Alives Discarded:            0
Total OAM Frames Received:              0
```

Enable-MPC

Use the **enable-mpc** command to dynamically start operation of a disabled MPC instance. When the MPC instance is enabled, existing configuration parameters are used, and the MPC statistics are not reset to their initial values. Use **create-mpc** to start an MPC instance using the configuration parameters saved in the SRAM and to reset all statistics.

Syntax:
enable-mpc

Enable-protocol

Use the **enable-protocol** command to dynamically enable the MPC over IP or IPX.

Syntax:

enable-protocol	ip
	ipx

Disable-protocol

Use the **disable-protocol** command to dynamically disable the MPC over IP or IPX.

Syntax:
disable-protocol ip
ipx

Disable-MPC

Use the **disable-mpc** command to dynamically deactivate the MPC instance. Once the MPC has been disabled, all packets follow the normal routed path and no shortcut data forwarding occurs.

Syntax:
disable-mpc

Create-MPC

Use **create-mpc** to start an MPC instance using the configuration parameters saved in the SRAM and to reset all statistics.

Syntax:
create-mpc

Delete-MPC

Use the **delete-mpc** command to delete an existing MPC instance. The MPC ceases operation immediately.

Syntax:
delete-mpc

Clear-statistics

Use the **clear-statistics** command to reset all the statistics maintained for the MPC instance to their initial values.

Syntax:
clear-statistics

MPC Neighbor MPS Monitoring Commands

Enter the following commands at the `MPC-MPS>` command prompt.

Table 83. MPC Neighbor MPS Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
list	Displays a current list of all the MPSs that have been discovered by the MPC (all the MPSs for which the MPC may perform forwarding functions).

Table 83. MPC Neighbor MPS Monitoring Command Summary (continued)

Command	Function
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

List

Use the **list** command to produce a current list of all the MPSs for which the MPC may perform forwarding functions. The displayed information includes a list of MAC addresses for which the MPC is performing flow detection, the interface number of the LEC associated with each MAC address, and the control ATM address of the MPS.

Syntax:
list

Example:

```
MPC MPS>list
List of Neighbor MPSs for MPOA Client (36):
=================================================
  1) Control ATM: 39.84.0F.00.00.00.00.00.00.00.00.01.10.00.5A.01.A4.00.05

1 MAC Address(es) Learnt For This MPS:

  1) MAC Addr: x10.00.5A.01.A4.00   Associated LEC Intf #:  42
```

MPC VCC Monitoring Commands

Enter the following commands at the `MPC-VCC>` command prompt.

Table 84. MPC VCC Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
list	Displays all the VCCs that are currently associated with the MPC.
list-vcc	Displays detailed information about a particular MPC VCC.
delete-vcc	Deletes a VCC associated with the MPC.
vcc-statistics	Displays aggregated statistics related to all VCCs associated with the MPC, including VCCs that may no longer be active).
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

List

Use the **list** command to display all of the VCCs that are currently associated with the MPC. This display includes fully operational VCCs and those that are not completely operational.

Syntax:
list

Example:

```
MPC VCC>list
SVCs For MPC On ATM Interface 36 (total    2):
=============================================
    1) VPI/VCI 0/38    State: OPERATIONAL
```

```
Remote ATM: 39.84.0F.00.00.00.00.00.00.00.00.01.10.00.5A.01.A4.00.05
    2) VPI/VCI 0/39    State: OPERATIONAL
Remote ATM: 39.84.0F.00.00.00.00.00.00.00.00.03.10.00.5A.01.9A.00.04
```

List-VCC

Use the **list-vcc** command to display detailed information about a particular MPC VCC.

Syntax:
list-vcc *vpi vci*

Example:

```
MPC VCC>list-v
VPI, Range 0..255 [0]?
VCI, Range 0..65535 [0]? 39

  VPI/VCI: 0/39     State: OPERATIONAL    Calling Party: FALSE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:4388/4388
  Remote ATM Addr: 39.84.0F.00.00.00.00.00.00.00.00.03.10.00.5A.01.9A.00.04
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: LLC 1483
  H/W Path Valid: FALSE    Ref. Frame Cnt: 4810
  Frames Tx/Rx: 2754/2754     Bytes Tx/Rx: 275400/275400


(Direct) Shortcut Routes Using This VCC:
----------------------------------------
     1) Address/Mask: 3.4.1.8/255.255.255.255    Shortcut State: RESOLVED
```

Delete-VCC

Use the **delete-vcc** command to delete a VCC associated with the MPC. ATM signalling closes the VCC. Because of on-going traffic, the VCC may be re-established shortly after deletion, giving the appearance that it was never deleted.

Syntax:
delete-vcc *vpi vci*

VCC-Statistics

Use the **vcc-statistics** command to display aggregated statistics related to all VCCs associated with the MPC, including VCCs that may no longer be active.

Syntax:
vcc-statistics

Example:

```
MPC VCC>vcc
VCC Statistics For This MPC:
----------------------------
Total Call Setup Failures:              0
Total Incoming Calls Rejected:          0
Total Connections Released Locally:     0
Total Calls Placed Successfully:        1
Total Calls Received Successfully:      1
Total Remote Hangups (Normal):          0
Total Remote Hangups (Error):           0
```

MPC Ingress Cache Monitoring Commands

Enter the following commands at the MPC-Ingress> command prompt.

Table 85. MPC Ingress Cache Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
list	Displays all the IP entries in the MPC’s ingress cache.
list-ipx	Displays all the IPX entries in the MPC’s ingress cache. This command may be shortened to **xlist**.
list-entries	Displays detailed information about specific IP ingress cache entries.
list-entries-ipx	Displays detailed information about specific IPX ingress cache entries. This command may be shortened to **xshow-entries**.
delete-entries	Deletes specified IP ingress cache entries.
delete-entries-ipx	Deletes specified IPX ingress cache entries. This command may be shortened to **xdelete-entries**.
ingress-statistics	Displays aggregated statistics for all of the MPC’s ingress cache entries.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

List

Use the **list** command to display a list of all the IP entries in the MPC ingress cache. Use **list-entries** to display more detailed information.

Syntax:
list

Example:

```
MPC INGRESS>list
Ingress Cache For MPC on ATM Interface 36
=========================================


Ingress Cache Entries for Direct Host Routes:
---------------------------------------------
1)  Protocol Address: 3.4.1.8    Shortcut State: RESOLVED

Ingress Cache Entries for Direct Network Routes:
------------------------------------------------

Ingress Cache Entries for Derived Host Routes:
----------------------------------------------
```

List-ipx

Use the **list-ipx** command to display a list of all the IPX entries in the MPC ingress cache. Use **list-entries-ipx** to display more detailed information.

This command may be shortened to **xlist**.

Syntax:
list-ipx

Example:

```
MPC INGRESS>list-ipx
Ingress Cache For MPC on ATM Interface 36
=========================================


Ingress Cache Entries for Direct Host Routes:
---------------------------------------------

Ingress Cache Entries for Direct Network Routes:
------------------------------------------------
 1) Network Number (in hex): 35508661   Shortcut State: RESOLVED

Ingress Cache Entries for Derived Host Routes:
----------------------------------------------
1) Network Number/Host Number (in hex): 35508661/00.00.00.00.00.01  Shortcut State:
RESOLVED
Derived From: 35508661
```

List-entries

Use **list-entries** to display more detailed information about IP entries.

You will be prompted for a destination IP address and address mask. Detailed information is displayed for all entries in the MPC's ingress cache which match the specified address/mask combination. The information displayed for each destination includes:

- State of the entry
- Destination ATM address (if resolved)
- Statistics on frames sent to the MPS and those sent over the shortcut
- Remaining age values
- MTU of the destination
- Shortcut VCC being used for this entry, if any, and the type of encapsulation being used on the VCC
- Whether this is a local shortcut
- Data link layer header information returned in the resolution reply
- Control ATM address of the MPS which provided the address resolution

Syntax:
list-entries *destination-protocol-address mask*

Example:

```
MPC INGRESS>list-en
Destination Protocol Address [0.0.0.0]? 3.4.1.8
Destination Protocol Address Mask [255.255.255.255]?
Host Route Entries matching 3.4.1.8/255.255.255.255
---------------------------


Direct Host Routes :

1) Address: 3.4.1.8  Shortcut State: RESOLVED
Hold Down Cause: N/A   CIE Code: x0
Dest ATM: 39.84.0F.00.00.00.00.00.00.00.00.00.03.10.00.5A.01.9A.00.04
Remaining Age (mins:secs): 3:12      Last Request ID: xB
Destn MTU: 4376       Encaps. Type: TAGGED
LANE Encaps. Hdr: xN/A
Tag Value: x1
Shortcut VCC (VPI/VCI):    0/  39   Local Shortcut ?: FALSE
MPS: 39.84.0F.00.00.00.00.00.00.00.00.00.01.10.00.5A.01.A4.00.05
```

```
Derived Host Routes :

Network Route Entries matching 3.4.1.8/255.255.255.255
------------------------------

None found!
```

List-entries-ipx

Use **list-entries-ipx** to display more detailed information about IPX entries.

You will be prompted for a destination network number and destination node number. Detailed information is displayed for all entries in the MPC's ingress cache which match the specified network number/node number combination. The information displayed for each destination includes:

- State of the entry
- Destination ATM address (if resolved)
- Remaining age values
- MTU of the destination
- Shortcut VCC being used for this entry, if any, and the type of encapsulation being used on the VCC
- Whether this is a local shortcut
- Data link layer header information returned in the resolution reply
- Control ATM address of the MPS which provided the address resolution.

This command may be shortened to **xshow-entries**.

Syntax:
list-entries-ipx *destination-network-number destination-node-number*

Example:

```
MPC INGRESS>list-entries-ipx
Destination Network Number (in 8-digit hex) (1 - FFFFFFFE) [0]? 35508661
 Destination Node Number (in hex) (0x000000000000 for network destination):[00.00.00.00.00.00]?

Host Route Entries matching 35508661/000000000000
------------------------------

      Direct Host Routes :

      Derived Host Routes:

1) Network Number (in hex) 35508661  Shortcut State:  RESOLVED
      Hold Down Cause: N/A  CIE Code:  x0
      Dest ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.12.34.12.34.12.34.03
           Remaining Age  (mins:secs) : 17:17    Last Request ID: x0
      Destn MTU:  4381       Encaps. Type: TR 802.2-IPX-LANE
      LANE Encaps. Hdr: x0000004008005a6c3b778004ac47390d06a000110020
      Tag Value: N/A
      Shortcut VCC (VPI/VCI):    0/ 211   Local Shortcut ?: FALSE
      MPS: 39.99.99.99.99.99.99.00.00.99.99.01.01.00.04.AC.47.39.06.06

   Network Route Entries matching 35508661
   ------------------------------

   1) Network Number (in hex) 35508661  Shortcut State:  RESOLVED
      Hold Down Cause: N/A  CIE Code:  x0
      Destn: 39.99.99.99.99.99.99.00.00.99.99.01.01.12.34.12.34.12.34.03
           Remaining Age  (mins:secs) : 17:17    Last Request ID: x0
```

```
Destn MTU:  4381       Encaps. Type: TR 802.2-IPX-LANE
LANE Encaps. Hdr: x0000004008005a6c3b778004ac47390d06a000110020
Tag Value: N/A
Shortcut VCC (VPI/VCI):    0/ 211   Local Shortcut ?: FALSE
MPS: 39.99.99.99.99.99.99.00.00.99.99.01.01.00.04.AC.47.39.06.06
```

Delete-entries

Use the **delete-entries** command to delete specific IP ingress cache entries.

Syntax:
delete-entries*destination-protocol-address mask*

You will be prompted for a destination protocol address and address mask. All ingress cache entries which match this address/mask combination are then deleted.

Note: Because of ongoing traffic, an ingress cache entry for a particular destination may immediately get recreated after it has been deleted using this command, giving the appearance that the entry had not been deleted.

Delete-entries-ipx

Use the **delete-entries-ipx** command to delete specific IPX ingress cache entries.

This command may be shortened to **xdelete-entries**.

Syntax:
delete-entries-ipx *destination-network-number destination-node-number*

You will be prompted for a destination network number and destination node number. All ingress cache entries which match this network number/node number combination are then deleted.

Note: Because of ongoing traffic, an ingress cache entry for a particular destination may immediately get recreated after it has been deleted using this command, giving the appearance that the entry had not been deleted.

Ingress-statistics

Use the **ingress-statistics** command to display aggregated statistics for all the MPC's ingress cache entries.

Syntax:
ingress-statistics

Example:

```
MPC INGRESS>ingress
Ingress MPC Statistics For This MPC:
-----------------------------------
Total Resolution Requests Sent:         14
Total Refresh Res. Requests Sent:       13
Total Res. Rqst Retransmissions:        1
Total Res. Rqst Timeouts:               0
Total Res. Reply Successes:             14
Total Res. Reply NAKs:                  0
Total Res. Replies Discarded:           0
Total MPS Purges Recvd:                 0
Total MPS Purged Mappings:              0
Total MPS Purges Discarded:             0
```

```
Total Triggers Recvd:                  0
Total Triggers Discarded:              0
Total Keep Alives Recvd:               443
Total Inactive Mappings Deleted:       0
Total Frames Forwarded On Shortcuts:   4414
Total Data Plane Purges Recvd:         0
Total Data Plane Purged Mappings:      0
Total Data Plane Purges Discarded:     0
Total NHRP Purge Replies Transmitted:  0
```

MPC Egress Cache Monitoring Commands

Enter the following commands at the MPC-Egress> command prompt.

Table 86. MPC Egress Cache Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.
list	Displays all the IP entries in the MPC’s egress cache.
list-ipx	Displays all the IPX entries in the MPC’s egress cache. This command may be shortened to **xlist**.
list-entries	Displays detailed information about specific IP egress cache entries.
list-entries-ipx	Displays detailed information about specific IPX egress cache entries. This command may be shortened to **xshow-entries**.
purge-entries	Purges specified IP egress cache entries.
purge-entries-ipx	Purges specified egress IPX cache entries. This command may be shortened to **xpurge-entries**.
egress-statistics	Displays aggregated statistics for all of the MPC’s egress cache entries.
Exit	Returns you to the previous command level. See “Exiting a Lower Level Environment” on page 11.

List

Use the **list** command to display a list of all the IP entries in the MPC egress cache. Use **list-entries** to display more detailed information.

Syntax:
list

Example:

```
MPC EGRESS>list
Egress Cache For MPC on ATM Interface 36
========================================

Egress Cache Entries w/ MPOA-Tag Encapsulation:
-----------------------------------------------

    1) Protocol Address/Mask: 5.4.1.5/255.255.255.255     State:     ACTIVE


Egress Cache Entries w/ Native 1483 Encapsulation (Host Routes):
----------------------------------------------------------------


Egress Cache Entries w/ Native 1483 Encapsulation (Netwk Routes):
-----------------------------------------------------------------
```

List-ipx

Use the **list-ipx** command to display a list of all the IPX entries in the MPC egress cache. Use **list-entries-ipx** to display more detailed information.

This command may be shortened to **xlist**.

Syntax:
list-ipx

Example:

```
MPC EGRESS>list-ipx
Egress Cache For MPC on ATM Interface 36
========================================


Egress Cache Entries w/ MPOA-Tag Encapsulation:
-----------------------------------------

Egress Cache Entries w/ Native 1483 Encapsulation (Host Routes):
-----------------------------------------------
    1) Net/Host Num (in hex): 3/00.00.00.01.A2.00 State :   Active

Egress Cache Entries w/Native 1483 Encapsulation (Netwk Routes):
-----------------------------------------------
```

List-entries

Use **list-entries** to display detailed information about all the IP entries in the MPC egress cache.

You will be prompted for a destination protocol address and mask. Detailed information is then displayed for all egress cache entries which match this address/mask combination. The information includes:

- ATM addresses of the source and the imposing MPS
- Type of the entry
- Identity of the egress LEC corresponding to the imposition request
- Cache ID of the entry
- Its remaining age
- Statistics on received data packets
- Tag value, if applicable
- Data link layer header information
- Information on the different types of LANE extensions returned in the last MPOA Cache Imposition reply for this entry

Syntax:
list-entries *destination-protocol-address mask*

Example:

```
MPC EGRESS>list-en
Destination Protocol Address [0.0.0.0]? 5.4.1.5
Destination Protocol Address Mask [255.255.255.255]?

      Egress Cache Entries matching 5.4.1.5/255.255.255.255 :

 1) Address/Mask: 5.4.1.5/255.255.255.255   Entry Type: TAG
    LEC #:  2    Cache ID: x1   State: ACTIVE
```

```
MPS: 39.84.0F.00.00.00.00.00.00.00.00.00.01.10.00.5A.01.A4.00.05
Source: 39.84.0F.00.00.00.00.00.00.00.00.00.03.10.00.5A.01.9A.00.04
Remaining Age (mins:secs): 13:37
Recvd Octets: 463900
Recvd Frames Forwarded: 4639
Recvd Frames Discarded: 0
Tag Value: x1       Local Shortcut: FALSE
DLL Header: x0040000000019f0090005a01a40006a08b884b40aaaa030000000800
LANE Extensions in last Imposition reply:  None
```

List-entries-ipx

Use **list-entries-ipx** to display more detailed information about IPX entries.

You will be prompted for a destination network number and destination node number. Detailed information is displayed for all entries in the MPC's egress cache which match the specified network number/node number combination. The information displayed for each destination includes:

- State of the entry
- Destination ATM address (if resolved)
- Statistics on frames sent to the MPS and those sent over the shortcut
- Remaining age values
- MTU of the destination
- Shortcut VCC being used for this entry, if any, and the type of encapsulation being used on the VCC
- Whether this is a local shortcut
- Data link layer header information returned in the resolution reply
- Control ATM address of the MPS which provided the address resolution.

This command may be shortened to **xshow-entries** or **xs**.

Syntax:
list-entries-ipx *destination-network-number destination-node-number*

Example:

```
MPC EGRESS>list-entries-ipx
Destination Network Number (in 8-digit hex) (1 - FFFFFFFE) [0]? 3
Destination Node Number (in hex) (0x000000000000 for network destination):[00.00.00.00.00.00]?

Egress Cache Entries matching 3/000000000000

1) IPX Net/Host Num: 3/00000001a200 Entry Type: 1483 (Host,Direct)
Lec#: 1  Cache IP: x1  State: ACTIVE
MPS: 39.84.0F.00.00.00.00.00.00.00.00.00.04.10.00.5A.01.AC.00.05
Source: 39.84.0F.00.00.00.00.00.00.00.00.00.024.10.00.5A.01.9C.00.03
Remaining Age (mins:secs): 5:5
Recvd Octets: N/A
Recvd Frames Forwarded: N/A
Recvd Frames Discarded: N/A
Tage Value: N/A    Local Shortcut: FALSE
DLL Header: x0040000000001a20090005a00999906a00a2a0a10e0e003
LANE Extensions in last Imposition reply: Formats 7, 11, 13, 17
```

Purge-entries

Use the **purge-entries** command to purge specified IP egress cache entries.

You will be prompted for a destination protocol address and mask. All egress cache entries which match this address/mask combination are purged. This is done using the MPOA egress MPC-initiated egress cache purge request.

Note: Because of ongoing traffic, an egress cache entry for a destination may immediately get recreated after it has been purged, giving the appearance that the purge command may not have been successful.

Syntax:
purge-entries *destination-protocol-address mask*

Purge-entries-ipx

Use the **purge-entries-ipx** command to purge specified egress cache entries.

You will be prompted for a destination network number and destination node number. All egress cache entries which match this network number/node number combination are purged. This is done using the MPOA egress MPC initiated egress cache purge request.

Note: Because of ongoing traffic, an egress cache entry for a destination may immediately get recreated after it has been purged, giving the appearance that the purge command may not have been successful.

This command may be shortened to **xpurge-entries** or **xp**.

Syntax:
purge-entries *destination-network-number destination-node-number*

Egress-statistics

Use the **egress-statistics** command to display aggregated statistics for all the MPC's egress cache entries.

Syntax:
egress-statistics

Example:

```
MPC EGRESS>egr
Egress MPC Statistics For This MPC:
-----------------------------------
Total Imposition Requests Recvd:        18
Total Imposition Rqsts NAKed:           0
Total Imposition Updates Received:      17
Total Imposition Purges Received:       0
Total Imposition Purged Mappings:       0
Total E-MPC Purge Rqsts Sent To MPSs:   0
Total E-MPC Purge Rqst Retransmissions: 0
Total E-MPC Purge Rqst Timeouts:        0
Tot. Frames Recvd & Fwded (Software):   5510
Total Recvd Frames Discarded:           0
Total Data Plane Purge Rqsts Sent:      0
Total Data Plane Purge Rqst Retransmits:0
Total Data Plane Purge Rqst Timeouts:   0
Total Egress Cache Entries Aged Out:    0
```

MPC Configure Monitoring Commands

Enter the following commands at the `MPC-Configure>` command prompt to dynamically change values of the MPC configuration parameters. The changes occur immediately and are temporary. The changes are applied to the running MPC instance and are not saved.

The commands for setting various configuration parameters are very similar to the corresponding commands using `talk 6`. Under `talk 5`, however, there are separate explicit commands for setting each parameter that can be dynamically configured, while under `talk 6`, all parameters are configured under the **set** command.

Note: Dynamic enabling/ disabling or creation/deletion of an MPOA client can also be done and the commands are under the MPC-BASE menu.

Two additional sets of configuration parameters available using the `talk 5` **config** that do not have any `talk 6` counterparts are the **atm-packet-trace-filter** and **lan-packet-trace-filter** commands. You can configure a mask for an ATM address or MAC address respectively. MPOA client packets are only traced if the remote ATM address of the VCC on which the packet is being sent or received or the destination MAC address of the packet if on a legacy LAN interface matches the configured mask.

The first example will result in only those MPOA client packets being traced on the ATM interface that are being sent/received over VCCs to a remote ATM address with the last two bytes being X'05'.

Example:

```
MPC CONFIGURE>atm
    ATM Address Pkt Trace Filter Value (40 hex digits)
     [0000000000000000000000000000000000000000]?
     0000000000000000000000000000000000000005
    ATM Address Pkt Trace Filter Mask (40 hex digits)
     [0000000000000000000000000000000000000000]?
     00000000000000000000000000000000000000ff
```

The following example will result in MPOA client packet tracing on the legacy LAN interfaces of only those packets for which the destination MAC address has a ′40’ as the first byte.

Example:

```
MPC CONFIGURE>lan
    LAN Pkt Trace Filter Value (12 hex digits)
     [000000000000]? 400000000000
    LAN Pkt Trace Filter Mask (12 hex digits)
     [000000000000]? ff0000000000
```

See page 554 for a description of each command.

Table 87. MPC Configure Monitoring Command Summary

Command	Function
? (Help)	Displays all the commands available for this command level or lists the options for specific commands (if available). See “Getting Help” on page 10.

Table 87. MPC Configure Monitoring Command Summary (continued)

Command	Function
list	Displays the current values of all the dynamically configurable parameters.
reset	Resets all the dynamically configurable parameters to the values configured at the `talk 6` prompt. See "MPC Configuration Commands" on page 551 for more information.
pcr	Dynamically sets the peak cell rate for the MPC instance.
max-reserved-bandwidth	Dynamically sets the maximum reserved bandwidth for the MPC instance.
enable-lane-shortcuts	Dynamically enables LANE shortcuts for the MPC instance.
lane-shortcuts-src-mac	Dynamically sets the source MAC address to be used in frames transmitted on LANE shortcut VCCs.
configured-src-mac	Dynamically sets locally administered MAC address to be used as the source MAC address in frames transmitted on LANE shortcut VCCs..
frame-count	Dynamically sets the frame count used to control the rate of traffic required before the MPC will initiate a shortcut to a given protocol destination.
frame-time	Dynamically sets frame time used to control the rate of traffic required before the MPC will initiate a shortcut to a given protocol destination.
init-retry-time	Dynamically sets the value of the retry timer used to determine if a request is to be retried when there is no response in a specified amount of time.
max-retry-time	Dynamically sets the maximum value of the retry timer used to determine if a request is to be retried when there is no response in a specified amount of time.
hold-down-time	Dynamically sets the minimum time to wait before reinitiating a failed resolution attempt.
vcc-timeout	Dynamically sets the time after which VCCs will be cleared when there has been no activity.
accept-config-from-lecs	Dynamically specifies whether any configuration parameters received from the LECS will be accepted by the MPC.
fragmentation-mode	Dynamically controls the manner that the ingress MPC handles IP packet fragmentation.
atm-packet-trace-filter	Allows the user to restrict packet tracing to specific VCCs.
lan-packet-trace-filter	Allows the user to restrict packet tracing to and from LAN ports.
Exit	Returns you to the previous command level. See "Exiting a Lower Level Environment" on page 11.

MPC Packet Tracing

MPOA client packet traces can be activated from the Event Logging System (ELS) which is an integral part of the device operating system. See the chapter entitled "Using the Event Logging System" and the chapter entitled "Configuring and Monitoring the Event Logging System" in *8371 Interface Configuration and Software User's Guide* for additional information about ELS.

Note: Packet tracing for the MPOA server function is separate from that of the MPOA client function and is accessed as part of NHRP packet tracing.

For MPOA client packet tracing, use the MPOA ELS subsystem. MPOA client packet tracing supports the **set trace decode on** option. This option enables the MPOA output to be interpreted for viewing. For details on using the trace facility, see the description of the trace command in the chapter entitled "Configuring and Monitoring the Event Logging System" in *8371 Interface Configuration and Software User's Guide*

MPOA client packets are identified by three different events under the MPOA ELS subsystem.

- Event 61 traces all MPOA client control frames
- Event 62 traces all MPOA client data frames
- Event 63 traces all MPOA client frames on legacy LAN interfaces.

Sample Trace Output 1:

```
#1 Dir:INCOMING  Time:2.10.16.85  Trap:7611
Comp:MPOA  Type:UNKNOWN  Port:0  Circuit:0x000000  Size:245
----------------------------------------------------------------
** MPC MPOA/NHRP Frame on 1483 VCC **
AddressFamily:ATM_NSAP  ProtocolType:IPv4  HopCount:64  PacketSize:237
Checksum:0x6F02  ExtensionOffset:0x0044  Version:1
PktType:MpoaCacheImpositionR
equest
SrcAddrTL:20  SrcSubAddrTL:0  SrcProtoLen:4  DstProtoLen:4
ReqID:26
Src NBMA:39840F00000000000000000000310005A019A0004
Src Protocol Addr: 5.4.2.0  Dest Protocol Addr: 5.4.1.5
0040: 00 FF 00 00 11 18 03 C0  00 00 00 00 10 01 00 00    | ................ |
0050: 10 02 00 04 D0 00 00 5A  00 08 00 08 08 00 5A 00    | .......Z......Z. |
0060: 00 01 00 06 00 08 00 1C  08 00 5A 00 00 01 00 0A    | ..........Z..... |
0070: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    | ................ |
0080: 00 00 00 00 00 08 00 34  08 00 5A 00 00 01 00 0C    | .......4..Z..... |
0090: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    | ................ |
00A0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    | ................ |
00B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 08 00 08    | ................ |
00C0: 08 00 5A 00 00 01 00 10  90 00 00 25 00 00 00 02    | ..Z........%.... |
00D0: 00 00 00 00 1C 00 40 00  00 00 01 9F 00 90 00 5A    | ......@........Z |
00E0: 01 A4 00 06 20 4B 48 8B  80 AA AA 03 00 00 00 08    | .... KH......... |
00F0: 00 80 00 00 00
```

Sample Trace Output 2:

```
#3 Dir:OUTGOING  Time:2.10.16.85  Trap:7611
Comp:MPOA  Type:UNKNOWN  Port:0  Circuit:0x000000  Size:269
----------------------------------------------------------------
** MPC MPOA/NHRP Frame on 1483 VCC **
AddressFamily:ATM_NSAP  ProtocolType:IPv4  HopCount:255  PacketSize:261
Checksum:0x0DBE  ExtensionOffset:0x0058  Version:1
PktType:MpoaCacheImpositionR
eply
SrcAddrTL:20  SrcSubAddrTL:0  SrcProtoLen:4  DstProtoLen:4
ReqID:26
Src NBMA:39840F00000000000000000000310005A019A0004
Src Protocol Addr: 5.4.2.0  Dest Protocol Addr: 5.4.1.5
0040: 00 20 00 00 11 18 03 C0  14 00 00 FF 39 84 0F 00    | . ..........9... |
0050: 00 00 00 00 00 00 00 00  02 10 00 5A 01 9C 00 03    | ...........Z.... |
0060: 10 01 00 04 00 00 00 01  10 02 00 04 00 00 D0 00    | ................ |
0070: 00 08 00 08 08 00 5A 00  00 01 00 06 00 08 00 1C    | ......Z......... |
0080: 08 00 5A 00 00 01 00 0A  00 00 00 00 00 00 00 00    | ..Z............. |
0090: 00 00 00 00 00 00 00 00  00 00 00 00 00 08 00 34    | ................4 |
00A0: 08 00 5A 00 00 01 00 0C  00 00 00 00 00 00 00 00    | ..Z............. |
00B0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    | ................ |
00C0: 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00    | ................ |
00D0: 00 00 00 00 00 08 00 08  08 00 5A 00 00 01 00 10    | ..........Z..... |
00E0: 90 00 00 25 00 00 00 02  00 00 00 00 1C 00 40 00    | ...%..........@. |
00F0: 00 00 01 9F 00 90 00 5A  01 A4 00 06 20 4B 48 8B    | .......Z.... KH. |
```

Sample Configuration

Example Configuration of an MPOA client

Notes:

1 Enter the MPOA client configuration menus.

2 First add an MPC.

3 Confirm that an MPC has been added, then go into the config option for configuring this MPC.

4 List the current configuration. Since we just added the MPC, all the configuration parameters have been set to the default values.

4a Some of the parameters displayed cannot be configured but are displayed simply for convenience. For instance the interface indicated refers to the ATM interface number on which the MPC is being configured.

5 If we would like the MPC to be in disabled state on bringup, use the **disable** command. The MPC can be dynamically activated from the monitoring console (talk 5).

6 Use the **set** command to configure various MPC parameters. Check the list of parameters which can be set using this command as shown below.

7 The *frame-count* parameter can be configured as shown (setting it to 1 will effectively result in the MPC making shortcut attempts to every destination for which a packet is encountered, while setting it to a very large value will result in shortcuts only to destinations to which extremely heavy traffic is being sent).

8 Configure the ESI portion of the ATM address of the MPC (the MPC uses a single ATM address as both its control and data ATM address). Two ESIs have already been administered under the ATM interface configuration menus.

9 Configure the selector byte to be used along with the ESI as part of the MPC's ATM address.

10 Configure parameters related to the use of LAN emulation shortcuts.

Note: If LANE shortcuts are enabled, you are prompted for the choice of source MAC address to be used in the layer 2 header of LANE shortcut packets. Further, if option 2 (*locally configured MAC address* is chosen, then you are also prompted for the MAC address desired.

11 Confirm the new configuration using the **list** command.

```
  MOS Operator Console

For help using the Command Line Interface, press ESCAPE, then '?'

*t 6
Gateway user configuration
Config>p mpoa        1
Next Hop Resolution Protocol/Multi Protocol Over ATM user configuration
MPOA config>mpc 36

MPOA Client user configuration
MPC >?
ADD
LIST
CONFIG
REMOVE
EXIT

MPC >add         2

     MPC added on interface 36
MPC >list         3
    LIST OF CONFIGURED MPOA CLIENTS
       --------------------------------
```

```
       Interface                  Status
       ---------------------------------
          36                      ENABLED

MPC >config

MPC Configuration> ?
ENABLE
DISABLE
SET
LIST
EXIT


        MPC Configuration> list        4

MPC Configuration
-----------------
STATUS: ENABLED
Shortcut Setup Frame Count:       10    (frames)
Shortcut Setup Frame Time:        1     (sec)
Initial Retry Time:               5     (sec)
Maximum Retry Time:               40    (sec)
Hold Down Time:                   160   (sec)
VCC Timeout Period:               20    (min)
Accept Config From LECS:          Yes
Fragmentation Mode:               Maximize Shortcut Usage

Interface:                       36

ESI:                              Burned In ESI
Selector:                         0x 2
Desired PCR:                      155000 (kbps)
Maximum Reserved Bandwidth:       155000 (kbps)
Line Rate:                        155 (Mbps)
Enable LANE Shortcuts:            TRUE
Source MAC Address for Shortcuts: Burned In

 4a

MPC Configuration> disable         5
Disable MPC?  [Yes]?y
    MPC set to DISABLED

MPC Configuration> set ?       6
FRAME-COUNT (FOR SHORTCUTS)
FRAME-TIME (FOR SHORTCUTS)
INITIAL-RETRY-TIME
MAXIMUM-RETRY-TIME
HOLD-DOWN-TIME
VCC-TIMEOUT-PERIOD
ACCEPT-CONFIG-FROM-LECS
FRAGMENTATION-MODE
ESI
SELECTOR
PCR
MAX-RESERVED-BANDWIDTH
SHORTCUTS


MPC Configuration> set frame-count    7
Frame Count for Shortcut Setup (in frames):  [10]? 1

MPC Configuration> set esi       8
[1] Burned in ESI
[2] 12.34.56.78.9A.BC
```

```
[3] 12.12.12.12.12.12
 ESI:  [1]? 2

MPC Configuration> set selector      9
Selector Byte (in hex) [2]? 10
MPC Configuration> set short    10
Enable LANE Shortcuts?  [Yes]? y

  Choices for Source MAC Address for LANE Shortcuts:
   [1] Burned in ESI
   [2] Locally Configured MAC Address
   [3] MAC Address from the Resolution Reply

MAC Address Type for LANE Shortcuts:  [1]? 2
MAC Address for LANE Shortcuts:  [00.00.00.00.00.00]?42.42.42.42.42.42

MPC Configuration> list       11

MPC Configuration
-----------------
STATUS: DISABLED
Shortcut Setup Frame Count:       1     (frames)
Shortcut Setup Frame Time:        1     (sec)
Initial Retry Time:               5     (sec)
Maximum Retry Time:               40    (sec)
Hold Down Time:                   160   (sec)
VCC Timeout Period:               10    (min)
Accept Config From LECS:          Yes
Fragmentation Mode:               Maximize Shortcut Usage

Interface:                       36

ESI:                              12.34.56.78.9A.BC
Selector:                         0x10
Desired PCR:                      155000 (kbps)
Maximum Reserved Bandwidth:       155000 (kbps)
Line Rate:                        155 (Mbps)
Enable LANE Shortcuts:            TRUE
Source MAC Address for Shortcuts: Locally Configured
  42.42.42.42.42.42
```

Part 5. Appendixes

© Copyright IBM Corp. 1996, 1999

Appendix. Abbreviations

AAL ATM Adaptation Layer

AAL-5 ATM Adaptation Layer 5

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

ATM Asynchronous Transfer Mode

ATMARP
ARP in Classical IP

ATP AppleTalk Transaction Protocol

AUI attachment unit interface

AVI ATM virtual interface

ayt are you there

BAN Boundary Access Node

BBCM Bridging Broadcast Manager

BCM BroadCast Manager

BECN backward explicit congestion notification

BGP Border Gateway Protocol

BGP Border Growth Protocol

© Copyright IBM Corp. 1996, 1999

BNC bayonet Niell-Concelman

BNCP Bridging Network Control Protocol

BOOTP
BOOT protocol

BPDU bridge protocol data unit

bps bits per second

bandwidth reservation

BSD Berkeley software distribution

BTP BOOTP relay agent

BTU basic transmission unit

BUS Broadcast and Unknown Server

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

CIPC Classical IP Client

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 Local Area Network

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

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

IETF Internet Engineering Task Force

Ifc# interface number

IGP interior gateway protocol

ILMI Interim Local Management Interface

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

LE LAN Emulation

LEC LAN Emulation Client

LED light-emitting diode

LECS LAN Emulation Configuration Server

LES LAN Emulation Server

LES-BUS
LAN Emulation Server - Broadcast and Unknown Server

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

LSB least significant bit

LSI LANE 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

MSS Multiprotocol Switched Services

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

QoS Quality of Service

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

SDU Service Data Unit

SGID server group id

seqno sequence number

SGMP Simple Gateway Monitoring Protocol

SL serial line

SLIP Serial Line IP

SMP standby monitor present

SMTP Simple Mail Transfer Protocol

SNA Systems Network Architecture

SNAP Subnetwork Access Protocol
SubNetwork Attachment Point

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

SVN Switched Virtual Networking

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

TLV Type/Length/Value

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

UNI User-Network Interface

UTP unshielded twisted pair

VCC Virtual Channel connection

VINES VIrtual NEtworking System

VIR variable information rate

VL virtual link

VNI Virtual Network Interface

VR virtual route

WAN wide area network

WRS WAN restoral

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 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.

© Copyright IBM Corp. 1996, 1999

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.

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.

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

BCM. BroadCast Manager, an IBM extension to LAN Emulation designed to limit the effects of broadcast frames.

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).

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)

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)

Border Gateway Protocol (BGP). An Internet Protocol (IP) routing protocol used between domains and autonomous systems. Contrast with *Exterior Gateway Protocol (EGP)*.

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*.

BUS. Broadcast and Unknown Server, a LAN Emulation Service component responsible for the delivery of multicast and unknown unicast frames.

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)

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.

CIP. Classical IP.

CIPC. Classical IP Client.

Classical IP Client. A Classical IP component that represents users of the Logical IP Subnet.

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 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*.

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)

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.

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.

ESI. End System Identifier, a 6-byte component of an ATM address.

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. Contrast with *Border Gateway Protocol (BGP)*.

F

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.

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.

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.

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.

hysteresis. The amount the temperature must change past the set alert threshold before the alert condition is cleared.

I

I frame. Information frame.

IETF. Internet Engineering Task Force, an organization that produces Internet specifications.

ILMI. Interim Local Management Interface, SNMP-based procedures for managing the User-Network Interface (UNI).

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 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.

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.

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)*.

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 Client (LEC). A LAN Emulation component that represents users of the Emulated LAN.

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.

LEC. LAN Emulation Client.

LECS. LAN Emulation Configuration Server.

LES. LAN Emulation Server.

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)*.

LIS. Logical IP Subnet, an IP subnet implemented with ATM technology Virtual Networking (SVN) framework.

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 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.

MSS. Multiprotocol Switched Services, a component of IBM 's Switched Virtual Networking (SVN) framework.

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*.

Next Hop Resolution Protocol (NHRP). A routing protocol, specified in Internet Draft Version 10 which has been submitted for RFC status. The Next Hop Resolution Protocol defines a method for a source station to determine the Non-Broadcast Multi-Access (NBMA) address of the "NBMA next hop" towards a destination. The NBMA next hop may be the destination itself or the router in the NBMA network that is "nearest" to the destination. The source station can then establish an NBMA virtual circuit directly with the destination or the router and reduce the number of routing hops through the NBMA network.

network user address (NUA). In X.25 communications, the X.121 address containing up to 15 binary code digits.

NHRP. Next Hop Resolution Protocol

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*.

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 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.

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)

Quality of Service (QoS). The user-oriented performance of an end-to-end service which is accessed using performance parameters. In ATM networks, the following performance parameters determine the QoS of an end-to-end ATM connection: cell loss ratio, cell transfer delay, and cell delay variation.

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.

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)*.

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

SDU. Service Data Unit, data as it appears at the interface between a layer and the layer immediately above.

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.

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 IP 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).

SLIP. Serial Line IP, an IETF standard for running IP over serial communication links.

SNA management services (SNA/MS). The services provided to assist in management of SNA networks.

SNAP. (1) SubNetwork Access Protocol. (2) SubNetwork Attachment Point.

socket. An endpoint for communication between processes or application programs.

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 Attachment Point (SNAP). An LLC header extension that identifies the protocol type of a frame.

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)

SVN. Switched Virtual Networking, the name of IBM 's framework for building and managing switch-based networks.

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 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

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.

TLV. Type/Length/Value, a generalized information element in a LAN Emulation packet.

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.

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.

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. The Japanese version (J1) transmits 1.544 Mbps.

U

UNI. User-Network Interface, the interface between user equipment and an ATM switch network.

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.

VCC. Virtual Channel Connection, a connection between parties.

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 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 Local Area Network (VLAN). A logical grouping of one or more LANs based on protocol and subnet and used to isolate network traffic within these groups.

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

© Copyright IBM Corp. 1996, 1999

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

Readers' Comments — We'd Like to Hear from You

8371 Networking Multilayer Ethernet Switch
Software User's Guide
and Configuration Reference

Publication No. GC30-9688-01

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

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

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
GC30-9688-01

IBM®

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

IBM Corporation
Design & Information Development
Department CGF/Bldg. 656
PO Box 12195
Research Triangle Park, NC
27709-9990

Fold and Tape | **Please do not staple** | Fold and Tape

GC30-9688-01

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC30-9688-01

Spine information:

8371 Networking Multilayer
Ethernet Switch

8371 Interface Configuration