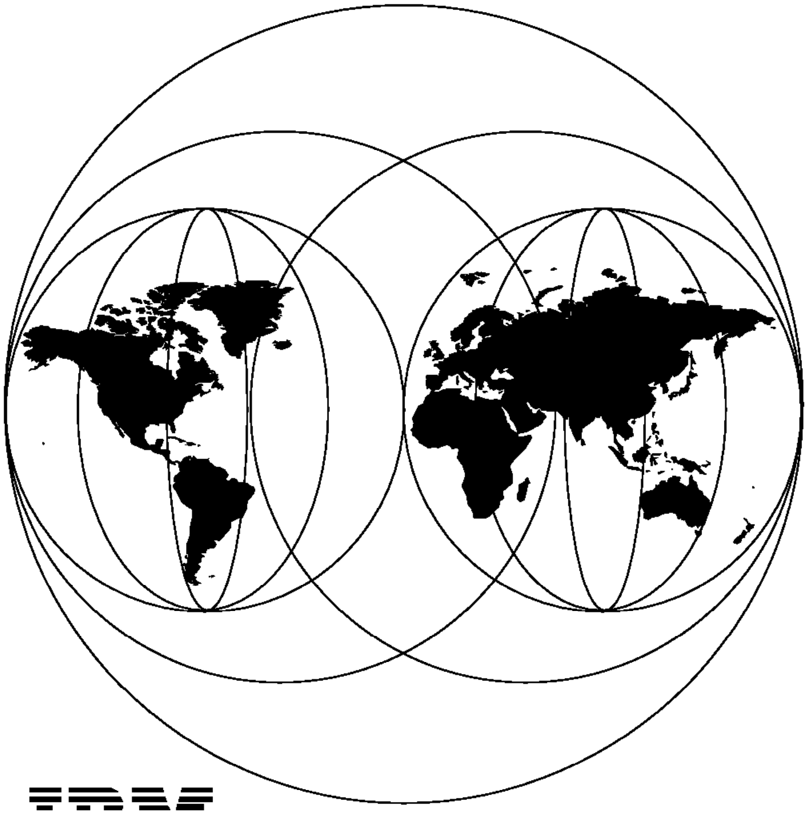
SG24-4915-00

Understanding and Using the IBM MSS Server

November 1996



IBM
International Technical Support Organization
Raleigh Center

International Technical Support Organization

SG24-4915-00

Understanding and Using the IBM MSS Server

November 1996

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix C, “Special Notices” on page 519.

First Edition (November 1996)

This edition applies to Release 1.0 (plus PTF) of the IBM 8210 Nways MSS Server.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 678
P.O. Box 12195
Research Triangle Park, NC 27709-2195

Contents

Preface

This redbook is intended to enable people to understand and use the functions offered by the IBM 8210 Nways MSS Server Release 1.0. Proper understanding of the technology, and following the many configuration examples given in this publication, will be helpful in designing and building an ATM network using IBM′s Multiprotocol Switching Services.

This publication is primarily intended for people involved in the design and construction of ATM networks used for data transport. Due to the many details given, it is also recommended for use by people responsible for maintenance and management.

Some knowledge of ATM is assumed.

How This Redbook Is Organized

This redbook contains 537 pages. It is organized as follows:

- Chapter 1, “MSS Server - Functional Overview”

  This chapter provides an introduction to the MSS Server. A description and an overview of the software functions is included.

- Chapter 2, “IBM MSS Server Configuration and Operation”

  This chapter provides an overview of the methods supported for configuring and accessing the MSS Server. We suggest that you read this chapter before starting to configure the MSS Server.

- Chapter 3, “MSS Server and ATM Ports”

  This chapter provides an overview of the configuration efforts required to connect the MSS Server to an ATM switch. A discussion is included on how VCC limitations on the MSS Server and the adjacent ATM switch may impact the connectivity.

- Chapter 4, “ATM Forum-Compliant LAN Emulation”

  The MSS Server provides extensive support for ATM Forum-compliant LAN emulation. Readers who are not familiar with its concepts are advised to read this section first.

- Chapter 5, “MSS Server and LAN Emulation”

  In addition to the basic ATM Forum-compliant (FC) LAN emulation support, the MSS Server introduces many enhancements. This chapter provides an overview of the IBM value-adds, and describes how to configure the basic and the valued-add functions.

  Note: Readers who are not familiar with FC LAN emulation are advised to read Chapter 4, “ATM Forum-Compliant LAN Emulation” on page 67 first.

- Chapter 6, “MSS Server and Classical IP”

  The MSS Server provides extensive support for Classical IP. This chapter provides an introduction to Classical IP, and describes how to configure Classical IP client and server functions on the MSS Server.

- Chapter 7, “MSS Server and IP Routing Protocols”

The MSS Server is capable of IP routing between logical IP subnets, between emulated LANs, and between logical IP subnets and emulated LANs. This chapter describe how to configure the MSS Server for IP routing. It also shows how to activate and configure the dynamic routing protocols.

- Chapter 8, “MSS Server and IPX Routing”

  The MSS Server is capable of IPX routing using LAN emulation and native ATM (RFC 1483) connections. This chapter gives an introduction to IPX, discusses the MSS Server′s IPX routing functions, and shows you how to configure them.

- Chapter 9, “MSS Server and Bridging”

  This MSS Server is capable of bridging between emulated LANs, supporting various types of bridging. This chapter gives a general introduction to bridging, and tells you how to use and configure them on the MSS Server.

- Chapter 10, “SNMP Management and Event Logging System (ELS)”

  The MSS Server provides SNMP support to enable management from remote SNMP network management stations. The MSS event logging system (ELS) is a powerful tool that can be used for monitoring purposes and for problem determination. This chapter gives you an overview of their functions and explains how to use and configure them.

- Chapter 11, “Implementation Scenarios”

  The MSS Server is a very versatile box, and confusion might result about how to configure its functions. This chapter contains a large number of simple implementation scenarios that can be used to build emulation LANs and/or Classical IP subnets. These scenarios can be used as basic building blocks for most complex environments, as is illustrated in the last, more complex, scenario.

- Appendix A, “ATM Forum-Compliant Frame Formats”

  This chapter provides you with the frame formats used in an ATM Forum LAN emulation environment. It is included for reference only.

- Appendix B, “Dynamic IP Routing Protocols - Introduction”

  This chapter gives an in-depth description of the dynamic IP routing protocols supported on the MSS Server. Readers who do not have experience with these protocols are advised to read this section first.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the Systems Management and Networking ITSO Center, Raleigh.

Volkert Kreuk is a Senior ITSO Specialist for 3745, 3746-9x0, NCP, and 8210 Nways MSS Server at the Systems Management and Networking ITSO Center, Raleigh. He writes extensively and teaches IBM classes worldwide on SNA/APPN, TCP/IP, frame relay, and ATM. Before joining the ITSO he worked in the IGN Technical Support group in the Netherlands.

Luke Gibbons is an Advisory Network Specialist at IBM Sydney, Australia. He has twelve years′ experience in the computer industry, seven of which have been in communications. He has a degree in Electronic Engineering and his main areas of expertise include ATM, bridging and routing and general network design.

Jaap de Goede is a technical consultant with Info′Products in the Netherlands, a business partner of IBM. He has eight years of experience in the LAN field, of which two years have been in the ATM environment. His major concentration is on designing, implementing and troubleshooting LANs. (E-mail: jdg@info-products.nl)

Carlos E. R. Pane is a Networking Systems Specialist in IBM Brazil. He has four years of experience in the LAN field. He holds a degree in Electrical Engineering. His areas of expertise include hubs, switches, bridges, routers and ATM, and his major work experience is on designing and implementing LANs.

Thanks to the following people for their invaluable contributions to this project:

Brian Elkins
International Technical Support Organization, Raleigh Center

John Parker
International Technical Support Organization, Raleigh Center

Aroldo Yai
International Technical Support Organization, Raleigh Center

Rene Castel
IBM France

MSS Server development
IBM USA

Special thanks to Ed Rovner of MSS Server development for explaining the many functions of the MSS Server and reviewing the redbook.

Comments Welcome

We want our redbooks to be as helpful as possible. Should you have any comments about this or other redbooks, please send us a note at the following address:

redbook@vnet.ibm.com

Your comments are important to us!

Chapter 1. MSS Server - Functional Overview

This chapter provides an introduction to the IBM NWAYS Multiprotocol Switched Services (MSS) Server, including a hardware description and an overview of the software functions. Further information can be found in *IBM Multiprotocol Switched Services (MSS) Server Configuration and Operations Guide* and *IBM Multiprotocol Switched Services (MSS) Introduction and Planning Guide*.

1.1 1996 Announcement

With the IBM Nways Multiprotocol Switched Services (MSS) Server attached to your campus asynchronous transfer mode (ATM) switch, you can form a high-performance multiprotocol backbone. This maximizes the effectiveness of your existing networks, while positioning your business for the demanding high-speed, low-delay applications of the future.

The Nways MSS Server provides ATM campuses with the following services:

- 155 Mbps multimode fiber or 155 Mbps single-mode fiber ATM connections to an ATM switch (stand-alone model)
- Supports UNI 3.0 and 3.1
- ATM Forum-compliant LAN emulation, including support for Ethernet emulated LANs and token-ring emulated LANs
- Virtual LANs, logical groupings of endstations
- Enhanced broadcast management support on emulated LANs for IP, Novell IPX, NetBIOS, and source-route bridge broadcasts
- Enhanced availability for LAN emulation by supporting redundant LAN emulation servers
- Transparent bridging support between Ethernet emulated LANs
- Source-route bridging support between token-ring emulated LANs
- Standards-based IP routing support on ATM, including support for Classical IP and routing between an emulated LAN and Classical IP
- Standards-based Novell IPX routing support on ATM between emulated LANs
- A user-friendly environment with:
  - A graphical configuration tool with integrated contextual help information
  - Streamlined configuration support running with a minimum of configuration input
  - A default configuration that includes a configuration needed for a test bed environment
  - HTTPD/HTML for command line monitoring and configuration using a Web browser
  - Integrated voice/fax modem (where homologated) that provides modem support and the ability to have faxes sent for reports or alerts to interact with a voice response unit to perform basic configuration, retrieve monitoring information, or to dial a pager in the event of a fault

1.1.1 Functional Overview

The IBM Nways MSS Server provides a multiprotocol networking solution for the ATM environment. It can be a LAN emulation server or a server for Classical IP over ATM.

The MSS Server provides ATM campuses with the following services:

- ATM Forum-compliant LAN emulation

  Allows ATM networks to appear as LANs to provide a migration path to ATM that protects investment in current LAN hardware and software. The IBM Nways MSS Server supports Ethernet and token-ring emulated LANs with up to 64 in each ATM interface. The Forum specifications for LAN emulation identify four components to implement an ELAN: LE clients (LEC), the LE Configuration Server (LECS), the LE Server (LES) and the broadcast and unknown server (BUS). The basic functions of these components are:

  - LAN emulation server - The LES provides address resolution and directory services to the LAN emulation clients on an ELAN. It provides a mapping of ATM address to MAC address to the requesting LEC. If the mapping is not registered then the LES will forward the request to the other LECs.
  - LAN emulation configuration server - The LECS assigns each LEC to the appropriate ELAN. According to the Forum specifications, it is not required to use the LECS to establish the link between the LEC and the LES. The LECS allows you to centralize the configuration and administration of your multiple ELANs and establish policies to connect LECs to LESs.
  - Broadcast and unknown server - The BUS handles frames sent to the broadcast MAC address, multicast frames and unicast frames that are not yet resolved or unknown by the LES.
  - LAN emulation client - The LEC provides an internal interface to higher layer protocols and emulates the MAC interface of an Ethernet or a token-ring LAN. This allows existing applications to use ATM services with LAN emulation. Therefore each workstation or device connecting to an ELAN must have an LEC. The LEC performs control function and data forwarding.

- Classical IP

  Allows IP networks to run over ATM without the use of LAN emulation by means of IP subnets created by an ARP Server. It supports Classical IP over ATM as specified in RFC 1577. RFCs 1755 for signaling and 1483 for packet encapsulation are also supported. The MSS Server provides IP routing between the logical IP subnets (LISs). It supports up to 32 LISs in each ATM interface.

- Virtual LAN Support

  Emulated LANs are not based on physical topology but, instead, on logical groupings of endstations. Having the stations logically grouped (ELANs or virtual IP subnet) allows much greater flexibility in handling moves, adds, or changes to the endstations.

- Enhanced LAN Emulation Functions

  In addition to ATM Forum-compliant LAN emulation, IBM offers several extensions:

  - Security

    The LAN emulation configuration server (LECS) can be used to check if the workstation attempting to join an ELAN belongs to that ELAN.

  - Redundancy

    The MSS Server can support one backup LAN emulation server on the same emulated LAN. This server stays in a backup state until it is needed. It can be activated if the original (primary) goes down in order to keep the network running.

  - Performance

    IBM extension BroadCast Manager (BCM) handles broadcast frames by sending them to an interested LAN emulation client. Reducing broadcasts reduces the traffic on the network and allows better performance and scalability.

  - Manageability

    Broadcast and unknown server (BUS) monitor is a function that provides a way to pinpoint end users who could be overutilizing the BUS. It periodically samples the BUS traffic on a particular ELAN identifying the top users of the BUS.

- Standards-based Bridging and Routing Support

  The IBM MSS Server supports the four commonly used bridging techniques (source-route bridging, transparent bridging, source route transparent bridging, and source-route to transparent bridging).

  Its extensive IP routing implementation includes OSPF, multicast support, and classless addressing in addition to such basic IP support as ICMP, UDP, TCP, ARP, and RIP.

  The MSS Server IPX routing support complies with the IPX Router Specification from Novell. IPX is supported over emulated LANs. Connections to other IPX routers are supported over ATM (RFC 1483).

1.1.2 Hardware Description

This product is available in two hardware platforms based on a PowerPC 603E 100MHz processor, including 512 KB of L2 cache, 12 MB of flash memory and 32 MB of DRAM:

1. The IBM 8210-001, which is a stand-alone product. It has two slots for 155 Mbps adapters (single-mode or multimode fiber) using SC connectors. It can be set on a surface or mounted in a standard 19-inch rack.
2. A module that is installed in the IBM 8260 Multiprocol Intelligent Hub. The MSS Server Module is based on the 8260 ATM Carrier Module and attaches to the ATM backplane of the 8260. It can be plugged into the 8260 Model A10 or A17. Note that each module used two slots and can be plugged into any slot except slot 11.

Both products, the stand-alone and the blade, also have:

- A PCMCIA hard disk (260 MB) for code and configuration storage (required in this release), which also can be used for first failure data capture of logging, trace and dump information.
- One standard service port that supports asynchronous communications for product configuration and maintenance. The port conforms to EIA 232 with a

male, 9-pin, D-shell connector and is capable of operating at up to 38.4 kbps. The port supports direct attachment or modem attachment and supports auto answer.

- A modem PCMCIA card (where the modem is homologated) for remote installation, network management, and service access to the module. The PCMCIA modem is a data/fax/voice modem in the USA and Canada, and a data/fax modem elsewhere.

Figure 1 and Figure 2 on page 5 show the front side of each type of MSS Server. As can be seen, the reset switch, all connectors and the LEDs are placed on the front.

Naming Convention

In terms of operational software, both hardware platforms are identical. The terms MSS Server and 8210 are used interchangeably throughout this publication, unless otherwise stated.

Required Code Level for the ATM CPSW

The MSS Server Module in the 8260 requires the ATM Control Point Switch Module (CPSW) at a code level 2.2.0 or higher, in order to assure that the 8260 recognizes the module and enables the LEDs.

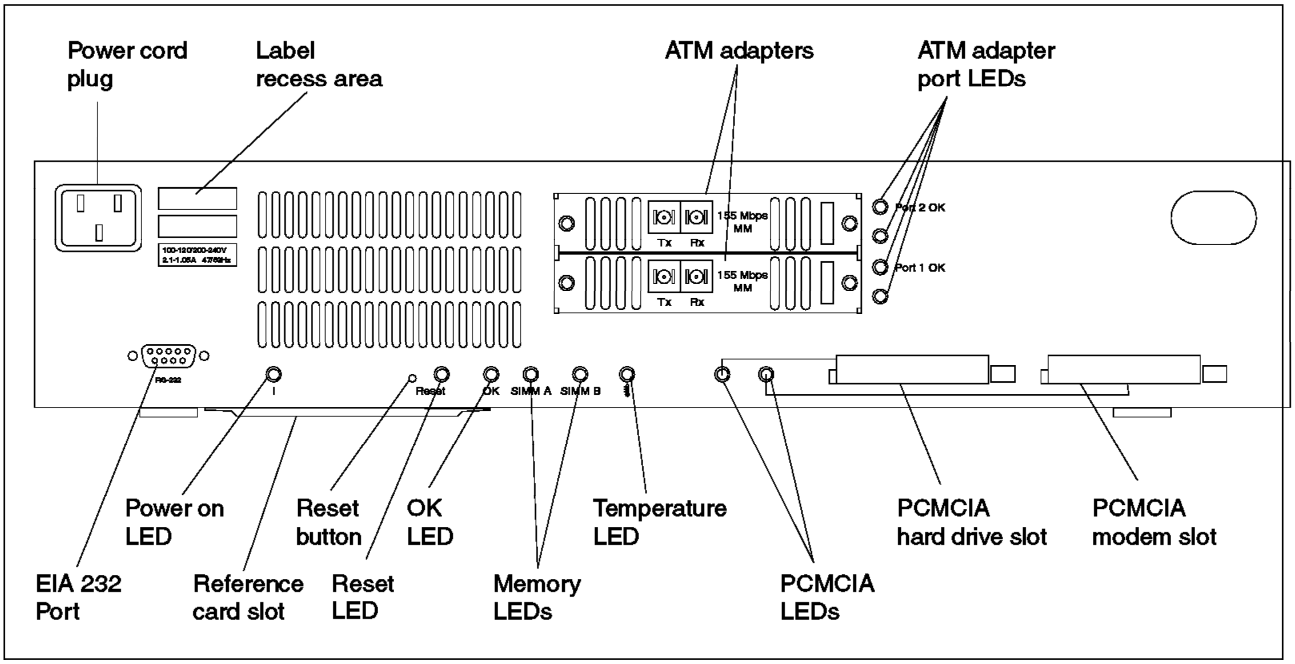


Figure 1. IBM 8210

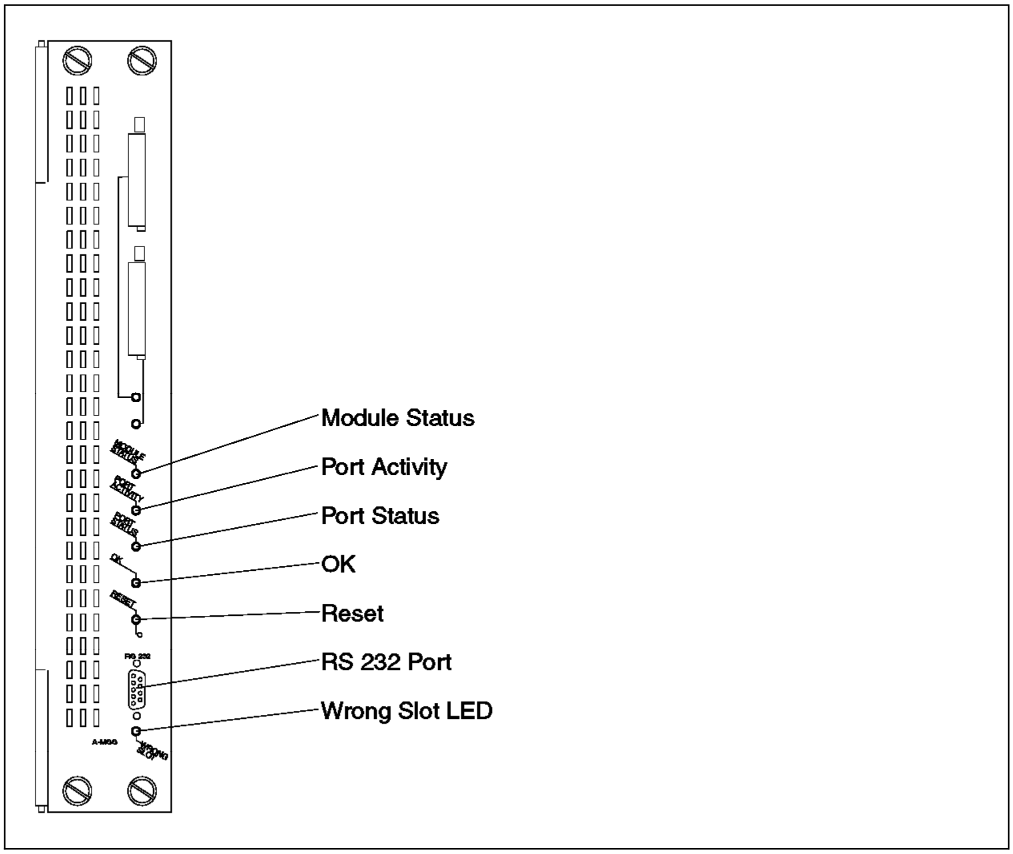


Figure 2. 8260 Module

Chapter 2. IBM MSS Server Configuration and Operation

This chapter describes the methods supported for configuring and accessing the IBM MSS Server. It also includes a description on how files are stored in the MSS Server and the boot process. Further information can be found in the *Multiprotocol Switched Services (MSS) Server Configuration and Operations Guide* and *Multiprotocol Switched Services (MSS) Server Command Line Interface Volumes 1* and *2*.

2.1 Methods of Connecting

Basically, you have three methods to connect to the MSS Server:

- Using a teletype (TTY) connection
- Using a serial line IP (SLIP) connection
- Using the ATM network

TTY and SLIP are considered *out-of-band* connections. Out-of-band connectivity is usually employed when the ATM network is not operational, or the 8210 has not been configured yet.

The connection via the ATM network is *in-band*. In-band connectivity requires IP connectivity over the 8210′s ATM network attachment. To enable in-band IP connectivity configure any of the following:

- LIS client (or server)
- LE client to which an IP address has been assigned
- IP host services

The in-band and out-band connectivity will be detailed in the following section.

2.1.1 Teletype (TTY) Attachment

For this method we have three alternatives:

- A local connection through a null modem cable attached to the EIA 232 service port (see Figure 3 on page 8)
- A remote connection through a modem attached to the EIA 232 service port (see Figure 4 on page 8)
- A remote connection through the PCMCIA modem (see Figure 5 on page 8)

You can set up a remote or a local connection, but only one port can be active at any given time. Either connection, local or remote, can be made using communications software, for example ProComm, that enables terminal emulation and file transfer.

The default settings for the service port are:

- Speed: 19.2 kbps
- Parity: None
- Data Bits: 8
- Stop Bits: 1

The PCMCIA modem is a 28.8-kbps V.32bis modem. Its default settings are the same as the service port, except the speed which is set to auto-detection.

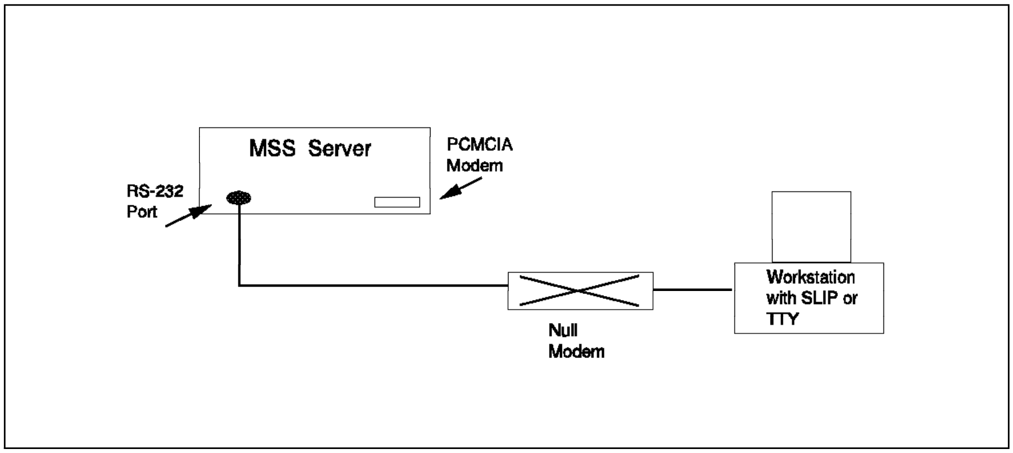


Figure 3. Local Connection Using the Service Port

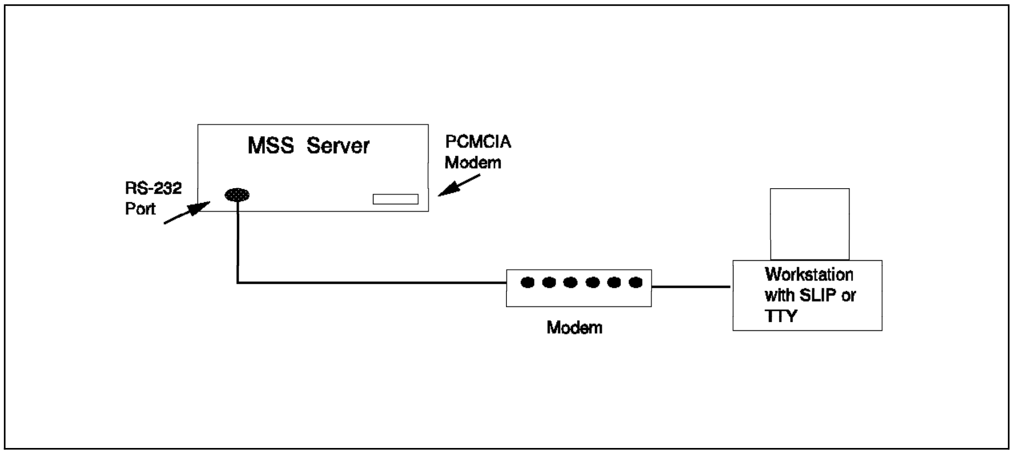


Figure 4. Remote Connection Using the Service Port

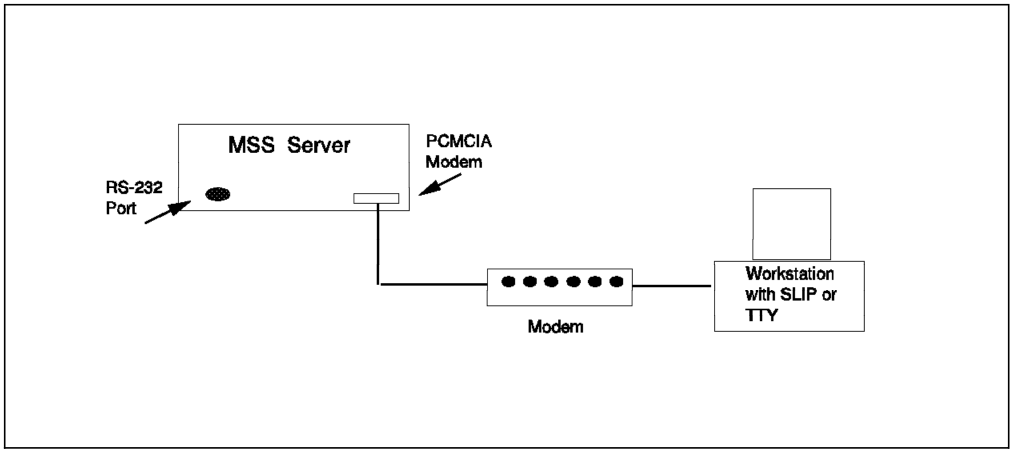


Figure 5. Remote Connection Using the PCMCIA Modem

2.1.2 SLIP

Over the local or remote connection mentioned earlier, you can choose to use the SLIP protocol instead of the TTY connection. Using SLIP requires TCP/IP on the workstation that connects to the MSS Server.

The default SLIP address of the MSS Server is:

10.1.1.2

You can use as your workstation SLIP address, for example:

10.1.1.3

To start a SLIP connection using TCP/IP for OS/2, enter:

```
mode com1: 19200,n,8
start /min slip -com1 -ifconfig 10.1.1.3 10.1.1.2 -mtu 1500 -speed 19200
```

Verify your connection with:

```
ping 10.1.1.2
```

For more information about installing SLIP, see the TCP/IP documentation related to your workstation.

2.1.3 ATM Network

You can only access the MSS Server via the ATM network after it has been configured. Therefore, for an intial configuration you have to use one of the two methods described in the previous two sections.

You can use quick config to set the IP address and the subnet mask for the ATM interface. See 2.3.3, “Quick Configuration” on page 14 for more information on quick configuration.

After you have completed the quick configuration, you have to reload the MSS Server to activate the configuration. After this reload the MSS Server is operational in the network and can be accessed via a LAN emulation or a Classical IP client. You can, for example, Telnet to the MSS Server or use the Web browser to access it.

2.2 Configuration Methods Supported

The MSS Server functions can be configured using any of the following tools:

- Configuration program

  The Configuration Program is the most user-friendly configuration method.

  It consists of a stand-alone software package for AIX, Windows, or OS/2 workstations to provide a graphical user interface for creating MSS configuration files. These files can be transferred to a MSS Server in the following ways:

  – Using Xmodem protocol to download the file over a TTY connection

  – Using trivial file transfer protocol (TFTP) to transfer the file over an SLIP or in-band IP connection

  - Using the Communications Option in the Configuration Program, which utilizes SNMP through a SLIP or in-band IP connection

  The Configuration Program runs on the following operating systems:

  - AIX (Version 3.2.5 or higher)
  - OS/2 (Version 2.1 or higher)
  - DOS/Windows (Windows 3.1 or higher, Windows 95 and Windows NT)

- Command line interface

  The command line console allows you to configure the MSS Server′s functions, as well as monitoring and providing status reports.

  It can be accessed using terminal emulation on the TTY connection, or by using Telnet over the SLIP or the in-band IP connection.

- Web browser HTML interface

  This interface brings the power of the command line interface through a more user-friendly graphical interface. The MSS Server supports the HyperText Transfer Protocol (HTTP) and HyperText Markup Language (HTML).

  The Web browsers can access the MSS Server via the SLIP or the in-band IP connection.

Each of these methods will be described in more detail in the following sections.

2.3 Command Line Interface

The MSS Server command line interface is accessible via terminal emulation on the TTY connection, or Telnet on the SLIP or the in-band IP connection.

When connecting to the MSS Server you will, by default, have access to all functions and commands. A configuration option exists to prompt users to enter a user ID and password when they connect. Enabling this function is described in 2.3.1, “Configuring User Access.”

2.3.1 Configuring User Access

To add controlled user access to the MSS Server, you need to enter the `Config>`**`add user`** command.

When you add a user, you are prompted for their password and their permission level, which can be one of the following:

- Administration

  This level gives access to all router commands and functions, including configuration and user administration.

- Operations

  This level allows you to view the configuration parameters and network statistics, use the GWCON commands to make dynamic configuration changes, and to restart the MSS Server.

- Monitor

  This level allows you only to view the configuration parameters and network statistics.

Note: User IDs can contain a maximum of eight characters, passwords can contain a maximum of eighty characters, and both user IDs and passwords are case sensitive.

When you add the very first user, the MSS Server automatically enables console login, which is the function that causes you to be prompted for a user ID and password. Console login can also be enabled and disabled using the following commands respectively:

```
Config>enable console-login

Config>disable console-login
```

Also, when you add the very first user, you are asked whether you wish to add technical support access. If you reply yes, an user ID called 2210l3 is created, with a password known by IBM service personnel. This user ID has administrator privileges and is available for use by IBM service personnel. However, you can change the password and permission level of this user if you wish, using the following command:

```
Config>change user
```

An example of adding users for the first time is shown in Figure 6 on page 12.

```
Config>add user
Enter user name:  []? admin
Password:                1
Enter password again:                 2
Enter permission: (A)dmin, (O)perations, or (M)onitor  [A]? a
User ′admin′ has been added
Enabling console login         3
Do you want to add Technical Support access?(Yes or [No]): y
Config>list user
   USER        PERMISSION
   admin       Admin
   221013      Tech Support
Console login is enabled

Config>add user hdesk   4
Password:
Enter password again:
Enter permission: (A)dmin, (O)perations, or (M)onitor  [A]? o
User ′hdesk′ has been added
Config>list user
   USER        PERMISSION
   admin       Admin
   221013      Tech Support
   hdesk       Operations
Console login is enabled

Config>add user Hdesk   4
Password:
Enter password again:
Enter permission: (A)dmin, (O)perations, or (M)onitor  [A]? o
User ′Hdesk′ has been added
Config>list user
   USER        PERMISSION
   admin       Admin
   221013      Tech Support
   hdesk       Operations
   Hdesk       Operations
Console login is enabled
```

Figure 6. Adding Users to the MSS Server

Notes:

1 The password field is hidden when entered.

2 You are required to verify the password.

3 Console login is automatically enabled.

4 An example to show that user IDs are case-sensitive.

2.3.2 MSS Server Process Structure

The user interface to the MSS Server consists of a main process, called OPCON, and several secondary processes that allow you to control and monitor the operation of the MSS Server.

The more commonly used processes and their structure are shown in Figure 7.

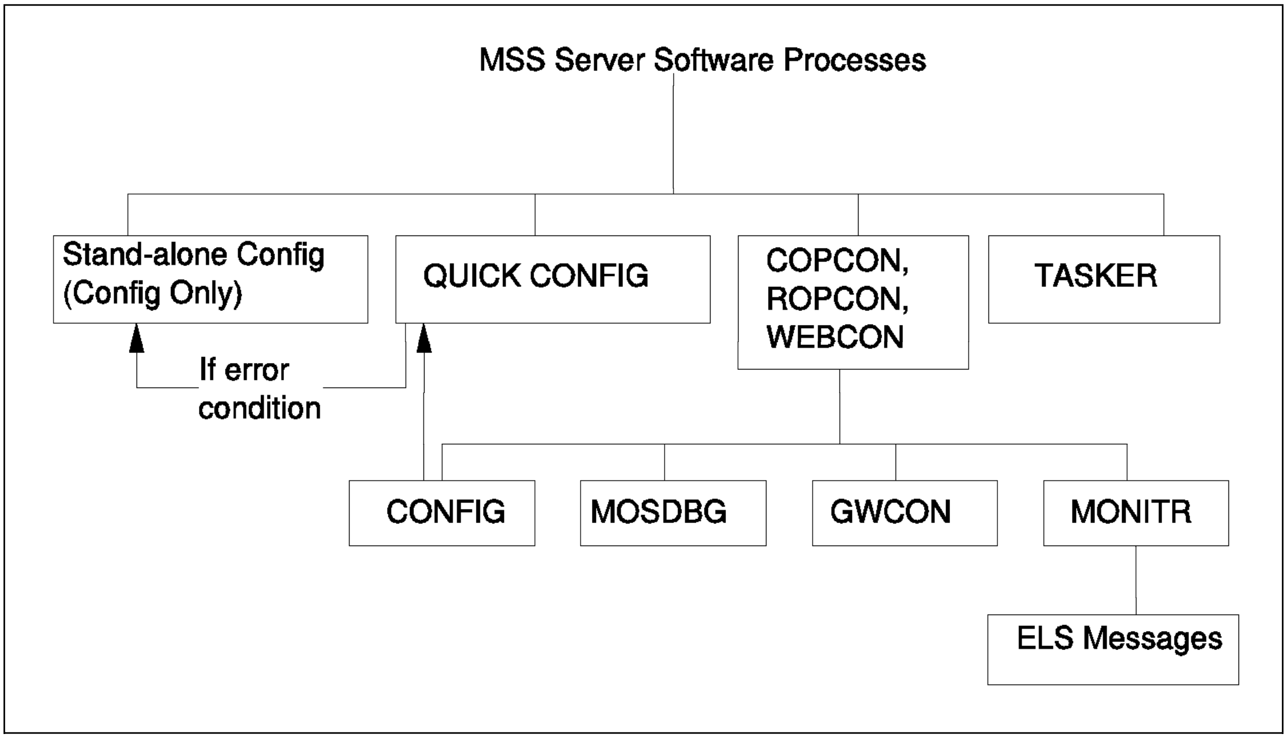


Figure 7. MSS Server Processes

Table 1 on page 14 gives a brief description of each process.

Table 1. Process Description

Process	**Description**	**Prompt**
OPCON	Functions as the main operator control program. It provides service for one directly connected console.	Asterisk (*)
ROPCON	Provides OPCON service for two remotely attached (Telnet) consoles.	Asterisk (*)
WEBCON	Provides OPCON service for a Web browser attached console.	None
GWCON	Provides monitoring of router′s hardware and software, protocols, network interfaces and event logging.	Plus sign (+)
CONFIG	Provides online configuration ability.	`Config>`
MONITR	Performs message printing services.	None
TASKER	Runs the MSS Server′s main networking software and performs the internetwork data transfer operations.	None
MOSDBG	Used as the Micro Operating System (MOS) runtime debugging tool.	Dollar sign ($)
Quick Config	Provides a simple way to configure devices, bridging and routing protocols and booting records.	None
CONFIG ONLY	Provides the same function as the CONFIG process with the addition of the reload command.	`Config (only)>`

2.3.3 Quick Configuration

The quick config process provides a quick and simple way to configure interfaces, bridging and routing protocols. Be aware, however, that this option assumes many defaults, some of which may not be appropriate for your installation. It is, therefore, advisable to proceed with a complete configuration after you reload the MSS Server. Quick configuration becomes available by issuing the `Config>`**`qconfig`** command.

These configurations will allow you to restart the MSS Server and configure it fully from a TCP/IP station, using the command line interface, the Configuration Program or the Web browser.

2.3.3.1 Examples

In this section we show some examples of using quick config. The examples are:

1. Define Classical IP client
2. Define Classical IP server (ARP server)
3. Define token-ring LAN emulation client

After each example we indicate the additional configuration required.

Note: Quick config does not enable you to change any of your ATM port definitions. See also 2.3.3.2, “Quick Config Considerations” on page 25.

```
Config>qconfig

Router Quick Configuration for the following:
o   LAN Emulation
        LAN Emulation Configuration Server (LECS)
        LAN Emulation Server (LES)
        LAN Emulation Client (LEC)
o   Bridging
        Spanning Tree Bridge (STB)
        Source Routing Bridge (SRB)
        Source Routing/Transparent Bridge (SR/TB)
o   Protocols
        IP (including OSPF, RIP and SNMP)
        IPX

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

Note:  Please be warned that any existing configuration for a particular item
will be removed if that item is configured through Quick Configuration

**********************************************************
LAN Emulation Configuration
**********************************************************

Type 'Yes' to Configure LAN Emulation
Type 'No' to skip LAN Emulation Configuration
Type 'Quit' to exit Quick Config
Configure LAN Emulation? (Yes, No, Quit): [ Yes] no  1

**********************************************************
Bridging Configuration
**********************************************************

Type 'Yes' to Configure Bridging
Type 'No' to skip Bridging Configuration
Type 'Quit' to exit Quick Config

Configure Bridging? (Yes, No, Quit): [ Yes] no

**********************************************************
Protocol Configuration
**********************************************************

Type 'Yes' to Configure Protocols
Type 'No' to skip Protocol Configuration
Type 'Quit' to exit Quick Config

Configure Protocols? (Yes, No, Quit): [ Yes] yes
Type 'r' any time at this level to restart Protocol Configuration

Configure IP? (Yes, No): [ Yes] yes
Type 'r' any time at this level to restart IP Configuration
```

Figure 8 (Part 1 of 3). An Example of Using Quick Config for Classical IP Client

```
Configuring Per-Interface IP Information

Configuring Interface 0 (CHARM ATM PCI Adapter)
Configure IP on this interface? (Yes, No): [ Yes] yes   2
IP Address: [ ] 192.168.20.10                           3
Address Mask: [ 255.255.255.0] 255.255.255.0            4
Is this an ATM ARP server? (Yes, No): [ Yes] no         5
Remote ATM ARP server address (40 hex digits starting with 39, 45, or 47):
[ ] 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.60.00.01.00  6

Per-Interface IP Configuration complete

Configuring IP Routing Information
Enable Dynamic Routing? (Yes, No): [ Yes] no

Only Static Routing Enabled

Routing Configuration Complete

SNMP will be configured with the following parameters:
     Community: public
     Access:    read_trap

This is the information you have entered:

     Interface #     IP Address         Address Mask
          0         192.168.20.10      255.255.255.0

Only STATIC Routing present.
Save this configuration? (Yes, No): [ Yes] yes


ATM Arp Clients:
--------------------------------------------------
If: 0  Prot: 0  Addr: 192.168.20.10    ESI: burned in          Sel: auto
Server: no   Refresh T/O: 5    AutoRefr: no    By InArp: yes  Validate PCR: no
Use Best Effort: yes/yes  (Control/Data)   Max B/W(kbps):      0
Cell Rate(kbps):  Peak:      0/     0    Sustained:      0/     0
Max SDU(bytes):   9188


ATM Arp Remote Server List:
   IP Address      Address / Sub Address
  192.168.20.10  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.60.00.01.00

IP configuration saved
```

Figure 8 (Part 2 of 3). An Example of Using Quick Config for Classical IP Client

```
Configure IPX? (Yes, No): [ Yes] no

Quick Config Done
Do you want to write this configuration? (Yes, No): [ Yes] yes
Config Save: Using bank B and config number 3, No): [ Yes] yes
(this line overwrites the previous one)


Configuration was written.
The system must be reloaded for this configuration to take effect.

Reload the system? (Yes, No): [ Yes] yes
```

Figure 8 (Part 3 of 3). An Example of Using Quick Config for Classical IP Client

Note:

1 In a Classical IP environment there is no need for LAN emulation.

2 Automatically enables Classical IP.

3 IP address for this client.

4 Must match the same subnet mask of the ARP server.

5 This is only an ARP client.

6 ATM address of the ARP Server of the logical IP subnet to which the client belongs.

After you reload the system, the MSS Server joins the logical IP subnet (LIS) and provides IP connectivity to it. Using the LIS client′s IP address, the MSS Server′s management functions can be accessed. An IP route statement is required if the management station is non-adjacent.

Telnet or Web browser access do not require additional configuration. If you intend to use the Configuration Program, you need to add some SNMP parameters. See 2.4.9.1, “MSS Server” on page 41 for more information.

```
Config>qconfig

Router Quick Configuration for the following:
o   LAN Emulation
        LAN Emulation Configuration Server (LECS)
        LAN Emulation Server (LES)
        LAN Emulation Client (LEC)
o   Bridging
        Spanning Tree Bridge (STB)
        Source Routing Bridge (SRB)
        Source Routing/Transparent Bridge (SR/TB)
o   Protocols
        IP (including OSPF, RIP and SNMP)
        IPX

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

Note:  Please be warned that any existing configuration for a particular item
will be removed if that item is configured through Quick Configuration

**********************************************************
LAN Emulation Configuration
**********************************************************

Type 'Yes' to Configure LAN Emulation
Type 'No' to skip LAN Emulation Configuration
Type 'Quit' to exit Quick Config

Configure LAN Emulation? (Yes, No, Quit): [ Yes] no      1

**********************************************************
Bridging Configuration
**********************************************************

Type 'Yes' to Configure Bridging
Type 'No' to skip Bridging Configuration
Type 'Quit' to exit Quick Config

Configure Bridging? (Yes, No, Quit): [ Yes] no

**********************************************************
Protocol Configuration
**********************************************************

Type 'Yes' to Configure Protocols
Type 'No' to skip Protocol Configuration
Type 'Quit' to exit Quick Config

Configure Protocols? (Yes, No, Quit): [ Yes] yes
Type 'r' any time at this level to restart Protocol Configuration
Configure IP? (Yes, No): [ Yes] yes
Type 'r' any time at this level to restart IP Configuration
```

Figure 9 (Part 1 of 2). An Example of Using Quick Config for Classical IP ARP Server

```
Configuring Per-Interface IP Information

Configuring Interface 0 (CHARM ATM PCI Adapter)
Configure IP on this interface? (Yes, No): [ Yes] yes  2
IP Address: [ ] 192.168.21.10
Address Mask: [ 255.255.255.0] 255.255.255.0
Is this an ATM ARP server? (Yes, No): [ Yes] yes       3
Selector byte (hex) (0 - FF): [ 0] 0        4

Per-Interface IP Configuration complete

Configuring IP Routing Information
Enable Dynamic Routing? (Yes, No): [ Yes] no

Only Static Routing Enabled

Routing Configuration Complete

SNMP will be configured with the following parameters:
     Community: public
     Access:    read_trap

This is the information you have entered:

     Interface #     IP Address         Address Mask
          0          192.168.21.10      255.255.255.0

Only STATIC Routing present.
Save this configuration? (Yes, No): [ Yes] yes


ATM Arp Clients:
--------------------------------------------------
If: 0  Prot: 0  Addr: 192.168.21.10    ESI: burned in          Sel: 00
Server: yes  Refresh T/O: 20   AutoRefr: yes   By InArp: yes  Validate PCR: no
Use Best Effort: yes/yes  (Control/Data)   Max B/W(kbps):      0
Cell Rate(kbps):  Peak:      0/     0    Sustained:      0/     0
Max SDU(bytes):   9188

IP configuration saved

Configure IPX? (Yes, No): [ Yes] no

Quick Config Done
Do you want to write this configuration? (Yes, No): [ Yes] yes
Config Save: Using bank B and config number 4, No): [ Yes] yes
(this line overwrites the previous one)


Configuration was written.
The system must be reloaded for this configuration to take effect.

Reload the system? (Yes, No): [ Yes] yes
```

Figure 9 (Part 2 of 2). An Example of Using Quick Config for Classical IP ARP Server

Note:

1 In a Classical IP environment, there is no need for LAN emulation.

2 Automatically enables Classical IP.

3 This is an ARP server.

4 This is the last byte of the ATM address of the ARP Server. We usually choose the default (0).

After you reload the MSS Server, it will become an ARP Server in the network. In order to access it, you have to configure client stations that belong to the same IP subnet and that have the ARP Server ATM address.

Quick config uses the *burned-in* address as the ESI part of the ATM address.

You can discover the ATM address using the GWCON prompt (+) as follows:

```
*
*talk 5


CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list address

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.00.04.13.47.39.36.00   1
ATM Interface+exit
ATM+exit
+    (press Ctrl+P to return to the CONFIG prompt)

*
```

Figure 10. How to Obtain the ATM Address

Note:

1 This is the ATM address of the ARP Server.

```
Config>
Config>qconfig

Router Quick Configuration for the following:
o   LAN Emulation
        LAN Emulation Configuration Server (LECS)
        LAN Emulation Server (LES)
        LAN Emulation Client (LEC)
o   Bridging
        Spanning Tree Bridge (STB)
        Source Routing Bridge (SRB)
        Source Routing/Transparent Bridge (SR/TB)
o   Protocols
        IP (including OSPF, RIP and SNMP)
        IPX

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

Note:  Please be warned that any existing configuration for a particular item
will be removed if that item is configured through Quick Configuration

**********************************************************
LAN Emulation Configuration
**********************************************************

Type 'Yes' to Configure LAN Emulation
Type 'No' to skip LAN Emulation Configuration
Type 'Quit' to exit Quick Config

Configure LAN Emulation? (Yes, No, Quit): [Yes] yes   1
Type 'r' any time at this level to restart LAN Emulation Configuration

Configure LECS to direct clients requesting unspecified ELAN type to:
     (Token Ring, Ethernet): [Token Ring] token ring

LAN Emulation configuration for ATM adapter in slot 1:

Do you want to add an emulated Token Ring? (Yes, No): [Yes] yes
Enter the ELAN Name:  [Token Ring ELAN 1]? TR1    2
Enter the selector: (2 - FF): [2] 2
Do you want to enable Broadcast Manger for IP? (Yes, No): [No] no
Do you want to enable Broadcast Manger for IPX? (Yes, No): [No] no
Do you want to enable Broadcast Manger for NetBIOS? (Yes, No): [No] no
Do you want to enable Broadcast Manger for Source Routing? (Yes, No): [No] no
Do you want to add an emulated Ethernet? (Yes, No): [Yes] no


Save this configuration? (Yes, No): [Yes] yes
This is all configured LAN Emulation information:
```

Figure 11 (Part 1 of 3). An Example of Using Quick Config for LAN Emulation

```
LECS Configuration:
LECS will be run on ATM adapter interface 0

Client requests that do not include an ELAN name will be
handled as follows:

Token Ring:   TR1
Ethernet:     Not Configured
Unspecified:  TR1


ATM Emulated LANs:
ATM adapter
Interface     Type          ELAN Name
-----------   ----          ---------
   0          Token Ring    TR1


ATM LAN Emulation Clients:
ATM adapter   LEC
Interface     Interface   Type         ELAN Name
-----------   ---------   ----         ---------
   0             1        Token Ring   TR1

Note:  All clients are configured to use the ATM adapter′s burned-in
ESI.  They are also configured to request their ELAN by name from the LECS.


LAN Emulation configuration information has been successfully written.

**********************************************************
Bridging Configuration
**********************************************************

Type ′Yes′ to Configure Bridging
Type ′No′ to skip Bridging Configuration
Type ′Quit′ to exit Quick Config

Configure Bridging? (Yes, No, Quit): [Yes] no

**********************************************************
Protocol Configuration
**********************************************************

Type ′Yes′ to Configure Protocols
Type ′No′ to skip Protocol Configuration
Type ′Quit′ to exit Quick Config

Configure Protocols? (Yes, No, Quit): [Yes] yes
Type ′r′ any time at this level to restart Protocol Configuration
```

Figure 11 (Part 2 of 3). An Example of Using Quick Config for LAN Emulation

```
Configure IP? (Yes, No): [Yes] yes
Type 'r' any time at this level to restart IP Configuration

Configuring Per-Interface IP Information

Configuring Interface 0 (CHARM ATM PCI Adapter)
Configure IP on this interface? (Yes, No): [Yes] no 3

Configuring Interface 1 (ATM Token Ring LAN Emulation)
Configure IP on this interface? (Yes, No): [Yes] yes
IP Address: [] 192.168.4.1
Address Mask: [255.255.255.0] 255.255.255.0

Per-Interface IP Configuration complete

Configuring IP Routing Information
Enable Dynamic Routing? (Yes, No): [Yes] yes
Enable OSPF? (Yes, No): [Yes] yes

OSPF Enabled with Max routes = 1000 and Max routers = 50

Routing Configuration Complete

SNMP will be configured with the following parameters:
     Community: public
     Access:    read_trap

This is the information you have entered:

     Interface #     IP Address         Address Mask
          1          192.168.4.1        255.255.255.0

OSPF is configured, and RIP is configured only for 'sending'.

Save this configuration? (Yes, No): [Yes] yes

IP configuration saved

Configure IPX? (Yes, No): [Yes] no

Quick Config Done
Do you want to write this configuration? (Yes, No): [Yes] yes
Config Save: Using bank B and config number 3, No): [Yes] yes
(this line overwrites the previous one)


Configuration was written.
The system must be reloaded for this configuration to take effect.

Reload the system? (Yes, No): [Yes] yes
```

Figure 11 (Part 3 of 3). An Example of Using Quick Config for LAN Emulation

Note:

1 We enable the LAN emulation configuration.

2 This is the ELAN name, and it is case-sensitive.

3 We do not need Classical IP.

Quick config uses the *burned-in* MAC address as ESI for the LECS, the LES-BUS and the internal LEC.

In order to have access to the MSS Server using an LEC, you must either define the LECS address into the switch or configure your LEC to not use the LECS and configure the LES/BUS address instead.

In order to obtain the ATM address of the LECS and the LES/BUS, you can use the MSS Server operator console.

```
*talk 5

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list address

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.03.00.04.13.47.26.51.02   1
39.09.85.11.11.11.11.11.11.11.11.01.03.00.04.13.47.26.51.00   2
ATM Interface+exit
ATM+exit
+    (press Ctrl+P to return to the CONFIG prompt)
*
```

Figure 12. ATM Network Prefix

Note:

1 The LES/BUS ATM address uses selector byte X′02′

2 The LECS ATM address uses selector byte X′00′

To verify your configuration, you can use the operator console to see if the internal client has joined the ELAN (see Figure 13 on page 25).

```
*
*talk 5


CGW Operator Console

+network 0
ATM Console
ATM+le-services
LE-Services Console
LE-SERVICES+work TR1
LE-Services Console for an existing LES-BUS Pair
EXISTING LES-BUS ′TR1′+database list all lec
Number of LEC′s to display: 1

    LEC-LES and LEC-BUS State  (UP=Up,  ID=Idle,  --. --.
      **=Other; Show specific LEC to see actual)    v   v
                                             LEC    State   #ATM  #Reg   #Lrnd
LEC Primary ATM Address               Proxy  ID    LES BUS  Adrs  MACs    MACs
----------------------------------------  -  ----   --  --  ----  -----  -----
3909851111111111111111010300041347265103  N  0001   UP  UP     1      1      0
EXISTING LES-BUS ′TR1′+

EXISTING LES-BUS ′TR1′+exit
LE-SERVICES+exit
ATM+exit
+    (press Ctrl+P to return to the CONFIG prompt)
*
```

Figure 13. LE Client Joined the ELAN

2.3.3.2 Quick Config Considerations

As mentioned in the beginning of this topic, quick config uses many default parameters, some of which might not be adequate for your installation.

The most important of these default parameters is the UNI version. Quick config does not change any ATM interface parameter and uses the default UNI Version 3.0. If the adjacent ATM switch has not been configured for UNI 3.0, no connection will be established.

Another important parameter is the ESI. Quick config uses the burned-in MAC address for its ESI. Especially for the ARP server, LECS, and LES/BUS, it is strongly recommended that you use a locally-administrated ESI. This will simplify administration and problem determination. If you need to change the UNI version or decide to use a locally-administrated ESI, see Chapter 3, “MSS Server and ATM Ports” on page 53 for an example.

2.3.4 CONFIG ONLY Process

The CONFIG ONLY process is used for making configuration changes without the normal operational router software running. In this state the MSS Server is not active in the network.

There are two ways to enter CONFIG ONLY mode:

- When no configuration file is active
- When the MSS Server encounters a problem during operation and automatically comes up in CONFIG ONLY mode

2.3.5 OPCON Process

The OPCON process handles the communication between the user and the other processes. To communicate with those other processes you need to issue the `talk pid` command from the OPCON prompt (*), where *pid* is the process ID of the process you wish to talk to. The process ID can be obtained by issuing the `status` command from the OPCON prompt.

Normally you can expect to see the following output:

```
*status
Pid  Name      Status TTY  Comments
1    COpCon    IOW    TTY0
2    Monitr    DET     --
3    Tasker    IDL     --
4    MOSDBG    DET     --
5    CGWCon    IOW     --
6    Config    IOW    TTY2
7    ROpCon    IDL    TTY1 9.24.104.110
8    ROpCon    IDL    TTY2
9    WEBCon    IDL     --
```

To return to the OPCON process from any other you need to use the *intercept character* which is set to Ctrl+P by default. To change the intercept character, issue the `intercept` command from the OPCON prompt.

The OPCON process also allows you to reload the MSS Server, and among other things, displays memory usage information. The commands available under OPCON are listed in Table 2.

Note

You can abbreviate most of the commands entered at any of the process prompts to the least number of letters that still make the command unique.

For example, the Talk command can be shortened to t.

Table 2 (Page 1 of 2). OPCON Commands

Command	Function
? (Help)	Lists all the OPCON commands.
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 intercept character. By default it is Ctrl+P.
Logout	Logs out the current session for the user who entered the command.
Memory	Reports the MSS Server′s memory usage.
Reload	Reloads the MSS Server software.

Table 2 (Page 2 of 2). OPCON Commands

Command	Function
Status	Shows information about all the MSS Server′s processes.
Talk	Connects to another MSS Server process and enables the use of its commands.
Telnet	Connects to another MSS Server, IP router, or to a remote host.

2.3.6 CONFIG Process

When using the command line or the Web browser interface, most of the configuration will be done using the CONFIG process. The CONFIG process is accessed by issuing the `*talk 6` command. CONFIG is fully described in the *Multiprotocol Switched Services (MSS) Server Command Line Interface Volumes 1* and *2*.

Using CONFIG commands, you can:

- Enter the quick configuration mode
- Enter the boot config mode
- Clear, list or update interface and protocol configuration information
- Enable or disable console login

The commands available under CONFIG are shown in Table 3.

Table 3 (Page 1 of 2). CONFIG Commands

Command	Function
? (Help)	Lists the CONFIG commands or lists the options associated with specific commands.
Add	Adds an interface or a user to the MSS Server′s configuration.
Boot	Enters boot config command mode.
Change	Changes the current interface′s device type configuration, or a user′s configuration/password.
Clear	Clears configuration information.
Delete	Deletes an interface or a user from the MSS Server′s configuration.
Disable	Disables console login, dump-memory, reboot system or a specified interface.
Enable	Enables login from a remote console or remote access using a modem, or enables a specified interface.
Event	Enters the ELS configuration command mode.
Feature	Provides access to configuration commands for specific router features outside protocol and interface. For example, the MCF (MAC Filtering).
List	Displays the system parameters or the hardware configuration.
Network	Enters the configuration environment of the specified network (interface).
Patch	Modifies the MSS Server′s global configuration.
Protocol	Enters the command environment of the specified protocol.
Qconfig	Enters the Quick Config process.
Set	Sets various system wide parameters
Time	Sets and displays the system time.

Table 3 (Page 2 of 2). CONFIG Commands

Command	**Function**
Unpatch	Restores patch variables to their default values.
Update	Updates the current version of the configuration.
Write	Stores a configuration file created using the command line interface.

Note: The many configuration scenarios we have included in Chapter 11, “Implementation Scenarios” on page 289 have all been defined using the CONFIG process.

> **Important**
>
> Changes to the configuration that are made using Config are written directly to the static random access memory (SRAM). The changes will take effect after a reload of the MSS Server. Unintentional changes can create havoc after the next reload. Verify the settings for any parameters you have configured before submitting them.

The structure of the configuration process for the MSS Server is shown in Figure 14 on page 29.

The command to enter the config process is `talk 6` from OPCON. The prompt changes into `Config>` and the following configuration commands become available:

- Boot

  Allows you to access the Boot config process (see 2.3.10, “Boot Config Process” on page 31). The prompt becomes `Boot Config>`.

- Event

  The new prompt becomes `ELS Config>` and configuration of the event logging system can start.

- Feature

  You are prompted for the MCF feature, the new prompt becomes `MCF Config>`, and configuration of MAC filtering can start.

- Network

  The network command enables you to configure the ATM physical interface or one of the logical interfaces associated with a token-ring or Ethernet LAN emulation client. You can either add the appropriate interface number to the network command, or enter it after being prompted by the configurator. Entering the ATM physical interface number provides also access to the configuration services for LE components.

- Protocol

  The protocol commands allow you to configure any of the protocols supported on the MSS Server:

  – IP

    After entering the `protocol IP` command you can configure the IP characteristics. The prompt changes into `IP Config>`.

  – ARP

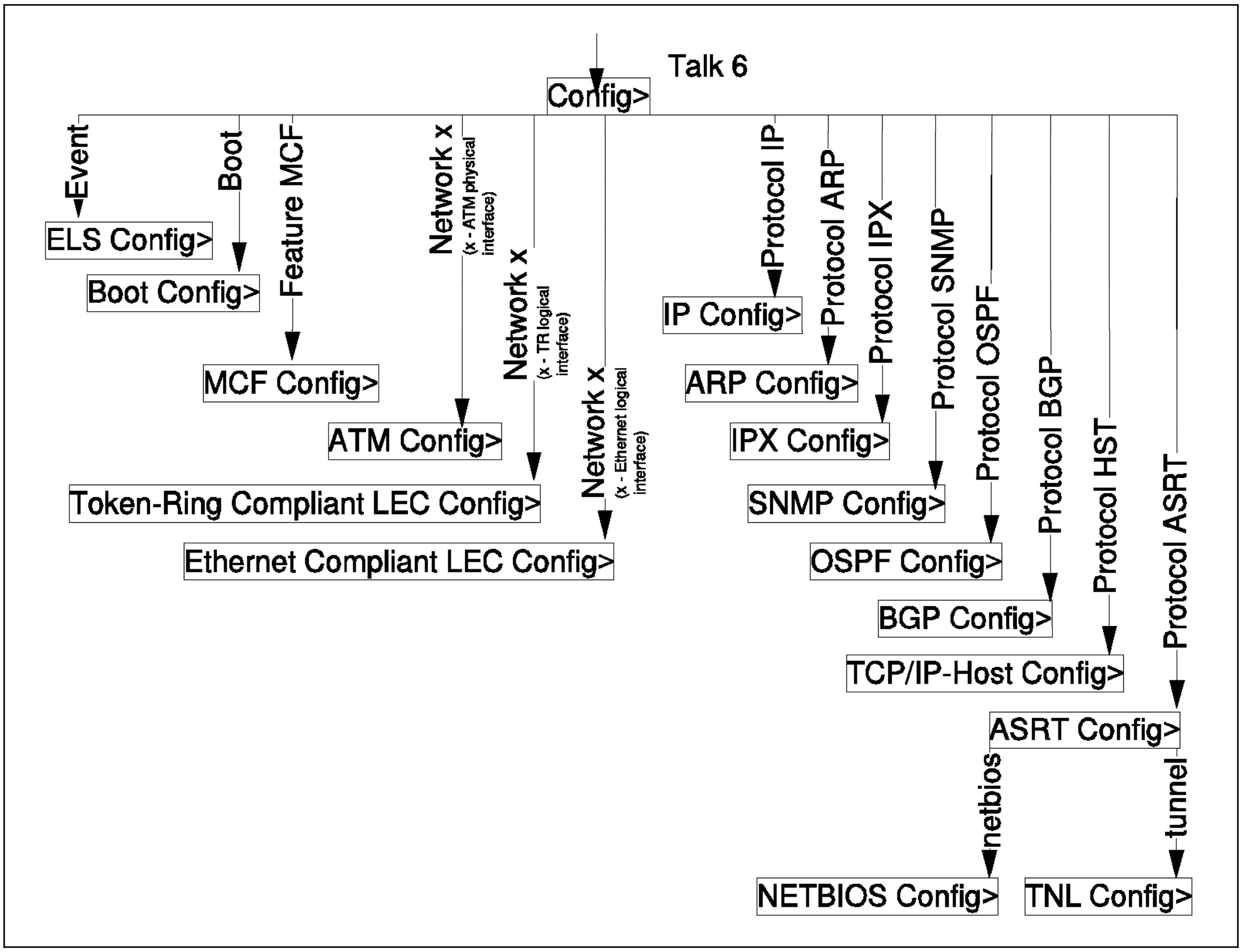


Figure 14. Config Structure

After entering the `protocol ARP` command you can configure the TCP/IP address resolution protocol (ARP) characteristics. It also provides access to the configuration services for ATMARP IPX (IPX using RFC 1483) and classical IP components. The prompt changes into `ARP Config>`.

– IPX

After entering the `protocol IPX` command you can configure the IPX characteristics. The prompt changes into `IPX Config>`.

– SNMP

After entering the `protocol SNMP` command you can configure the TCP/IP SNMP characteristics. The prompt changes into `SNMP Config>`.

– OSPF

After entering the `protocol OSPF` command you can configure the open SPF-based protocol (OSPF). The prompt changes into `OSPF Config>`.

– BGP

After entering the `protocol BGP` command you can configure the border gateway protocol (BGP). The prompt changes into `BGP Config>`.

– HST

After entering the protocol HST command you can configure the MSS Server′s IP host services. The prompt changes into HST Config>.

– ASRT

After entering the protocol ASRT command you can configure the adaptive source-routing transparent bridging (ASRT) protocol. The prompt changes into ASRT Config>.

To define bridge tunneling enter tunnel. The prompt changes into TNL Config>.

To define the NetBIOS bridging configuration enter netbios. The prompt changes into NETBIOS Config>.

2.3.7 GWCON Process

The GWCON process is accessed by issuing the talk 5 command from OPCON. You can use this process to monitor the current router configuration, monitor router utilization, and make temporary configuration changes.

Using GWCON commands you can:

- List the interfaces and protocols currently configured
- Display memory and network statistics
- Set some current configuration parameters
- Set current ELS parameters
- Test, enable or disable a network interface

The commands available under GWCON are shown in Table 4.

Table 4. GWCON Commands

Command	**Function**
? (Help)	Lists the GWCON commands.
Buffer	Displays information about packet buffers assigned to each interface.
Clear	Clears statistics.
Configuration	Lists the currently configured protocols and interfaces.
Disable	Disables a specified interface.
Error	Displays error statistics for the network.
Event	Enters the ELS event console.
Feature	Enters the console commands for specific features.
Interface	Displays statistical information about the network interfaces.
Log	Sets or displays the logging level for events not included in ELS.
Memory	Displays CPU memory usage, packet sizes, and number of buffers.
Network	Enters the console environment of the specified network (interface).
Protocol	Enters the console environment of the specified protocol.
Queue	Displays buffer statistics for a specified interface.
Statistics	Displays statistics for a specified interface.
Test	Enables a disabled interface or tests the specified interface.

2.3.8 MONITR Process

The MONITR process receives messages from the Event Logging System (ELS) and displays them on the console. It is accessed by issuing the `*talk 2` command.

There is no prompt associated with MONITR and you cannot enter any commands. To exit MONITR you need to use the OPCON intercept character (Ctrl+P by default).

2.3.9 MOSDBG Process

This is a special process used by software specialists to examine and change the contents of the MSS Server′s memory and registers. It is used only for debugging purposes and is not discussed here.

2.3.10 Boot Config Process

The Boot Config process is accessed by entering `boot` from the `Config>` prompt.

It allows you to:

- Add, modify or delete entries from the boot configuration database
- Enable or disable network memory dumping
- View the current boot and dump configuration database
- Transfer configuration information between router memory and remote hosts, using TFTP
- Retrieve the boot file from a local or remote source
- Store boot file to the available banks
- List the contents of the code image banks
- Delete files from the image banks

From the `Boot Config>` prompt, you have the following commands available:

Table 5. Boot Config Commands

Command	**Function**
? (Help)	Displays a list of the commands available from this prompt level.
Copy	Copy boot files and configuration within the MSS Server.
Describe	Displays information about the stored images in the image bank.
Erase	Erases a stored image or an image bank.
List	Displays software boot configuration.
Set	Defines the boot config file within the bank which contains the active image.
TFTP	Initiates TFTP file transfers between the MSS Server and local and remote hosts.
Exit	Leaves the Boot Config environment and returns to the CONFIG process.

2.4 MSS Configuration Program

The Configuration Program is available for the AIX, OS/2 or DOS/Windows operating systems. The minimum requirements for each operating system and the procedures for installing the MSS Configuration Program are listed in the following sections.

2.4.1 AIX

Requirement for AIX

Workstation	RS/6000 POWERstation or POWERserver
RAM	16 MB
Diskette Drive	1.44 MB, 3.5 inch
Free Hard Disk Space	14 MB
Display	Graphics display that supports 640x480 resolution and 16 colors or gray scale
Mouse	3-button
Software	AIX 3.2.5 or higher with TCP/IP enabled AIX Windows Environment/6000

The procedures for installing the software under AIX are:

1. Log in to AIX as a non-root user (make sure this user has write access to the directory you want to install the software).
2. Insert Program Disk 1 into the diskette drive.
3. Type `dosread -a INSTALL.AIX INSTALL.AIX` and press enter.
4. Type `chmod 550 INSTALL.AIX` and press enter.
5. Type `./INSTALL.AIX` and press enter.
6. Follow the installation program instructions.

To start the program, you must be in an X-Windows window.

1. Change to the directory that contains the Configuration Program files.
2. Type `./cfg`.

2.4.2 OS/2

Requirement for OS/2

Workstation	IBM PS/2 or other IBM-compatible-PC with Intel 80386 or higher processor.
RAM	15 MB (16 MB recommended)
Diskette Drive	1.44 MB, 3.5 inch
Free Hard Disk Space	14 MB
Swapper Size	10 MB
Display	Graphics display that supports 640x480 resolution and 16 colors or gray scale
Mouse	2-button (functionally equivalent to a 3-button mouse)
Software	OS/2 2.1 or higher including WARP IBM TCP/IP 1.2.1 for OS/2 or higher

The procedure for installing the software under OS/2 are:

1. Open an OS/2 command window.
2. Insert Program Disk 1 into the diskette drive.
3. Make the diskette drive your current directory.
4. Type `os2inst` and press enter.
5. Follow the installation program instructions.

The program creates an IBM MSS folder that contains the MSS configuration program icon.

To start the Configuration Program:

1. Open the MSS folder.
2. Double-click on the Configuration Program icon.

2.4.3 Windows

Requirement for DOS/Windows

Workstation	IBM PS/2 or other IBM-compatible-PC with Intel 80386 or higher processor.
RAM	15 MB
Diskette Drive	1.44 MB, 3.5 inch
Free Hard Disk Space	14 MB
Display	Graphics display that supports 640x480 resolution and 16 colors or gray scale
Mouse	2-button (functionally equivalent to a 3-button mouse)
Software	IBM DOS 5.0 or higher, or Microsoft DO S 5.0 or higher Microsoft Windows 3.1 or higher with a TCP/IP application that uses WinSock 2.0 Win32s, included with the Configuration Program WinSock 2.0 (included with Win32s)

The procedures for installing the software under DOS/Windows are:

1. Install the Win32s support if not currently installed. Win32 is not required with Windows 95.
2. Insert Program Disk 1 into the diskette drive.
3. Select **File** from the Program Manager menu.
4. Select **Run** from the File drop-down menu.
5. Enter *d*:`\win\install` in the dialog, where *d* is the drive letter of your diskette drive.
6. Follow the installation program instructions.

The program adds a program group named IBM MSS to the Program Manager. This group contain the MSS Configuration Program.

To start it:

1. Open the MSS group.
2. Double-click on the Configuration Program icon.

2.4.4 MSS Configuration Program - Overview

The MSS Configuration Program consists of two main windows:

- The Navigation Window
- The Configuration window

The Navigation Window displays a directory tree consisting of the various components that you can configure. To select any particular configuration screen, click the left mouse button on the item you are interested in. The Configuration window will now display the configuration screen you have selected.

Help is available for each field within a panel. You may access the help by pressing PF1.

If the field requires you to enter a value, make sure you press CR (Enter/Return) after entering your value. If you do not do this, the value may not be saved.

Specific configurations using the MSS Configuration Program are included in some examples shown in this book.

2.4.5 The Navigation Window

The Navigation Window lists the configuration screens grouped by function (for example, Devices, LE, LECS, ELANS, IP, IPX and so on). When you select a screen in the Navigation Window, the presentation window will display the configuration questions for the selected screen.

This Navigation Window is a method of presenting information hierarchically.

An example of a hierarchical structure is as follows:

```
Routers
  Devices
    General
```

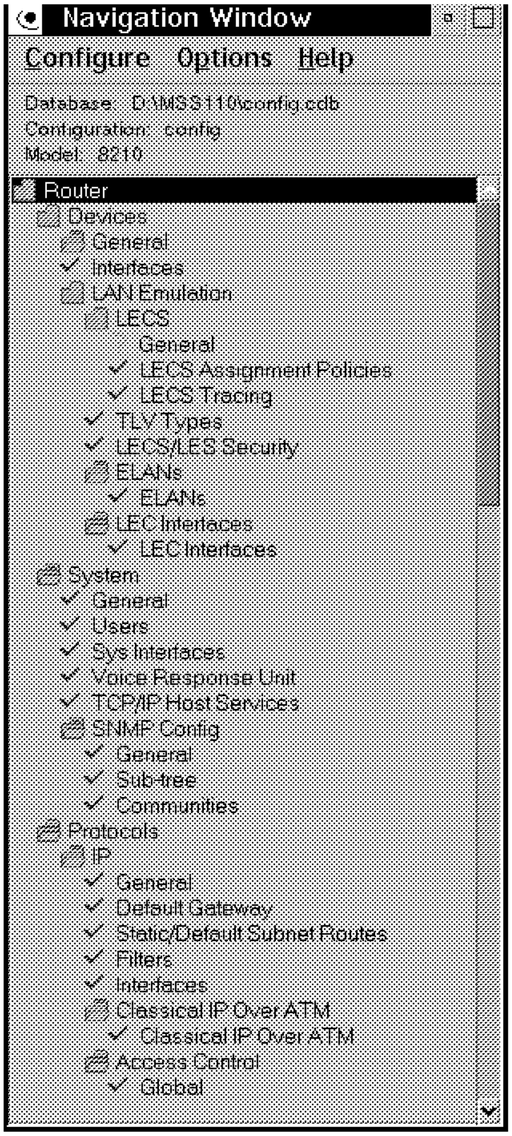
Note that folder icons are displayed by certain items in the navigation list. These items in the navigation tree group screens by function. Also note that there can be other nested folders within a given folder. The first time you start the Configuration Program the entire navigation list is presented. However, the navigation list can be contracted (and later on expanded) by selecting items that have folder icons.

The way to expand or contract the navigation list is by pressing and holding the middle button (on three-button mice) or the right button (on two-button mice) and selecting the appropriate choice from the displayed pop-up menu.

The Router category (at the top of the navigation list) is considered the root of the navigation tree. The next level of functions are as follows:

```
Router
  Devices
  System
  Protocols
  Bridging
  Features
```

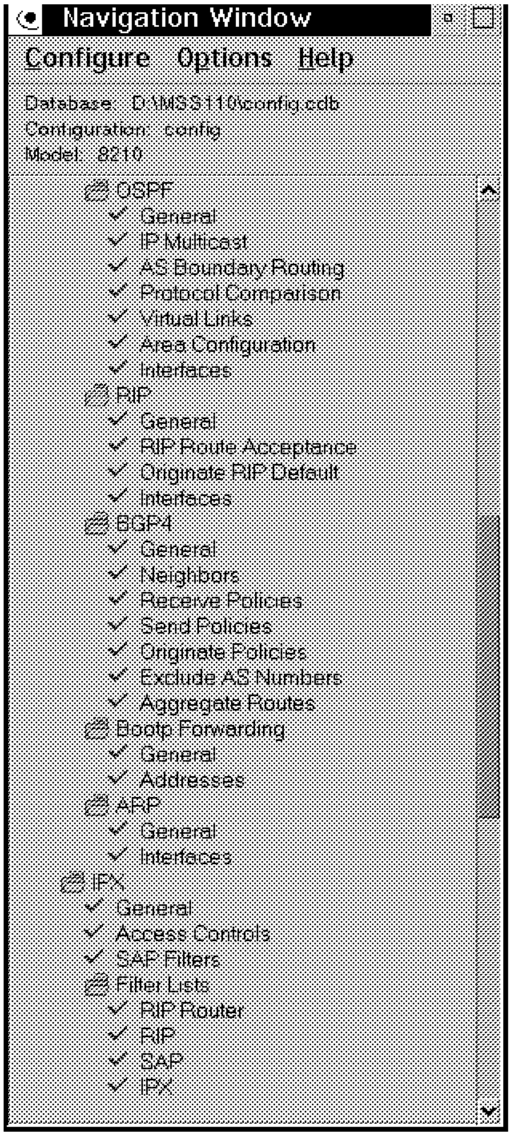
The complete Navigation Window is shown in Figure 15 on page 36.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

- Router
  - Devices
    - General
    - ✓ Interfaces
    - LAN Emulation
      - LECS
        - General
        - ✓ LECS Assignment Policies
        - ✓ LECS Tracing
      - ✓ TLV Types
      - ✓ LECS/LES Security
      - ELANs
        - ✓ ELANs
      - LEC Interfaces
        - ✓ LEC Interfaces
  - System
    - ✓ General
    - ✓ Users
    - ✓ Sys Interfaces
    - ✓ Voice Response Unit
    - ✓ TCP/IP Host Services
    - SNMP Config
      - ✓ General
      - ✓ Sub-tree
      - ✓ Communities
  - Protocols
    - IP
      - ✓ General
      - ✓ Default Gateway
      - ✓ Static/Default Subnet Routes
      - ✓ Filters
      - ✓ Interfaces
      - Classical IP Over ATM
        - ✓ Classical IP Over ATM
      - Access Control
        - ✓ Global

Figure 15 (Part 1 of 3). The MSS Configuration Program Navigation Window

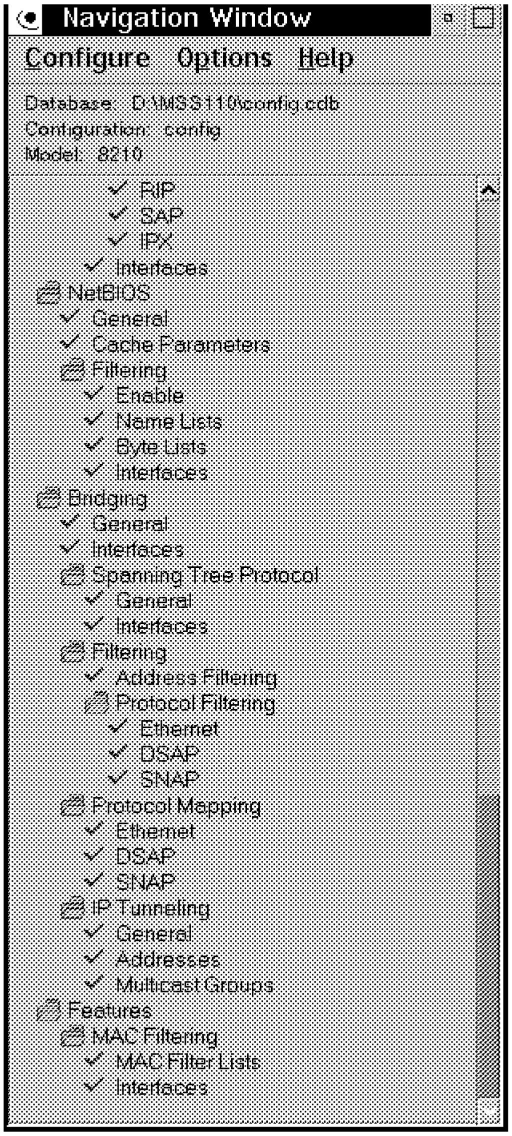


Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

- OSPF
  - ✓ General
  - ✓ IP Multicast
  - ✓ AS Boundary Routing
  - ✓ Protocol Comparison
  - ✓ Virtual Links
  - ✓ Area Configuration
  - ✓ Interfaces
- RIP
  - ✓ General
  - ✓ RIP Route Acceptance
  - ✓ Originate RIP Default
  - ✓ Interfaces
- BGP4
  - ✓ General
  - ✓ Neighbors
  - ✓ Receive Policies
  - ✓ Send Policies
  - ✓ Originate Policies
  - ✓ Exclude AS Numbers
  - ✓ Aggregate Routes
- Bootp Forwarding
  - ✓ General
  - ✓ Addresses
- ARP
  - ✓ General
  - ✓ Interfaces

- IPX
  - ✓ General
  - ✓ Access Controls
  - ✓ SAP Filters
  - Filter Lists
    - ✓ RIP Router
    - ✓ RIP
    - ✓ SAP
    - ✓ IPX

Figure 15 (Part 2 of 3). The MSS Configuration Program Navigation Window

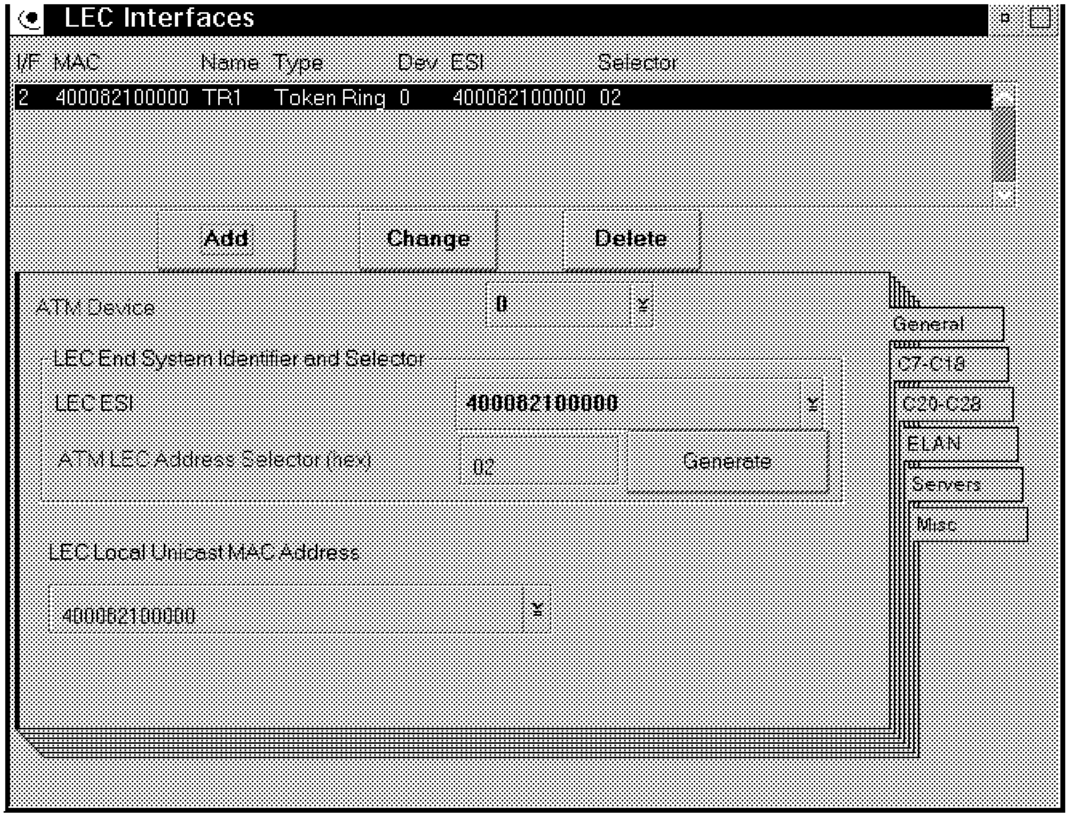


Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

```
        ✓ RIP
        ✓ SAP
        ✓ IPX
      ✓ Interfaces
  NetBIOS
    ✓ General
    ✓ Cache Parameters
    Filtering
      ✓ Enable
      ✓ Name Lists
      ✓ Byte Lists
      ✓ Interfaces
  Bridging
    ✓ General
    ✓ Interfaces
    Spanning Tree Protocol
      ✓ General
      ✓ Interfaces
    Filtering
      ✓ Address Filtering
      Protocol Filtering
        ✓ Ethernet
        ✓ DSAP
        ✓ SNAP
    Protocol Mapping
      ✓ Ethernet
      ✓ DSAP
      ✓ SNAP
    IP Tunneling
      ✓ General
      ✓ Addresses
      ✓ Multicast Groups
  Features
    MAC Filtering
      ✓ MAC Filter Lists
      ✓ Interfaces
```

Figure 15 (Part 3 of 3). The MSS Configuration Program Navigation Window

A sample panel is shown in Figure 16 on page 39. All panels within the Configuration Program will look similar to this one.



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100000	TR1	Token Ring	0	400082100000	02

ATM Device: 0

LEC End System Identifier and Selector

LEC ESI: 400082100000

ATM LEC Address Selector (hex): 02

LEC Local Unicast MAC Address: 400082100000

Figure 16. LEC Interfaces Configuration Screen

2.4.6 Configure Pull-Down Menu

The following options become available by clicking on the **Configure** button in the Navigation Window.

- New Configuration

  This option creates a new configuration. This means that you are starting a clean configuration from scratch for the MSS Server model (stand-alone or 8260 module) you choose from the submenu.

- Save Configuration As

  This option saves the current configuration in the configuration database. A single database can hold multiple configurations.

  Configurations saved in the Configuration Program are readable only by the Configuration Program. The MSS Server only understands configurations that are created by the Create Router Configuration option.

- Read Router Configuration

  This option loads a binary file into the Configuration Program. After loaded you can open it and save it.

- Create Router Configuration

  This option creates a configuration file which you may send to the router.

- Open Configuration

This option enables you to retrieve a configuration file from a configuration database that was previously saved by the Configuration Program.

- Delete Configuration

  This option enables you to delete a configuration from the configuration databases.

- Communications

  This option allows you to send or retrieve configurations to/from the MSS Server. You can send/retrieve a single configuration to/from the MSS Server by selecting the **Single Router** option, or you can send multiple configurations to multiple MSS Servers by selecting the **Multiple Routers** option. In this option you also can reload the MSS Server.

- ASCII File

  This option is for creating an ASCII version of the configuration file.

2.4.7 Options Pull-Down Menu (on the Navigation Window)

The following options become available by clicking on the **Options** button in the Navigation Window.

- Font

  This option allows you to select the fonts that are on your workstation.

- Color

  This option works by selecting a color with the three sliders representing the RGB values of the color. Once the desired color is displayed, click on either the **Warning** or **Error** color box to use that color.

- Message Prompting

  Message prompting filters out those types of messages which are turned on in the message prompting customization window. When the configuration is validated, only those classes of messages selected by the user will be displayed.

- Default Model

  This option allows you to set the default 8210 model (stand-alone or 8260 blade). On the next activation of the Configuration Program, the values associated with the model type become the initialized values.

- ATM Address Format

  This option allows you to choose the format that the ATM address will have (octet or subadress) and the separator it will use (none, colon(:), space() or dot(.)).

2.4.8 Validating Configuration Files

In the Navigation Window, check marks or question marks are displayed next to each screen or function name. Check marks indicate that the configuration data associated with the screen or function has been validated by the Configuration Program. Question marks indicate that one or more parameters are not valid.

Invalid configurations can be saved to database files, but you should not create router configuration files or attempt to send them to the MSS Server unless they have been succesfully validated by the Configuration Program.

2.4.9 Using the Configuration Option

Before you can use the Configuration option, IP connectivity must be present between the configuration workstation and the MSS Server.

First, an initial configuration must be loaded into the MSS Server. After that the SNMP protocol has to be configured on the MSS Server. Finally, the Configuration Program must be configured and enabled to access the MSS Server.

The following is an example of how to do it.

2.4.9.1 MSS Server

1. Access the MSS Server through one of the methods described previously in this chapter (TTY, Web browser, or Telnet).
2. Perform a quick configuration to define an IP address and IP subnet mask for the ATM interface. See 2.3.3, "Quick Configuration" on page 14 for details.
3. After the server has been reloaded, you have to set some SNMP parameters. Using the command line interface this is:
   a. Enter `talk 6` at the OPCON prompt (*).
   b. Enter `protocol snmp` at the `Config>` prompt.
   c. Define a community name with read and write access. For example:

```
SNMP Config>set community access write_read_trap
Community name
[]? my_community
Access set successfully
```

   Note: *my_community* is the SNMP community name defined. Replace it with the value appropriate for your configuration.

4. Exit SNMP configuration.
5. Exit Config.
6. Enter `reload` to restart the MSS Server.

Now the MSS Server is enabled for SNMP communication with the Configuration Program.

2.4.9.2 Configuration Program

To enable the configuration program to communicate with the MSS Server, you have to:

1. Select the appropriate model from the New Configuration menu.
2. Select **Interfaces** in the IP folder in the Navigation Window.
3. In the Configuration Window:
   a. Select **IP Addresses** next to the interface with which the Configuration Program will communicate.
   b. Enter the IP address and the Subnet mask for this interface.

      This address must match an IP address configured on the MSS Server.

   c. Select **Add**.
4. Select **Communities** under the SNMP folder in the Navigation Window.

5. Select **Read-write trap** in the drop-down list under Access type.
6. Select **Add**.

   Note: The name and access type must match the name and access type configured on the MSS Server.

Now you can communicate with the MSS Server and perform functions such as sending configuration and querying information.

2.5 Web Browser HTML Interface

The Web browser HTML/HTTP interface provides access to the MSS server′s configuration and console functions, diagnostics functions, and vital product data. By providing a graphical, full-screen interface you are able to maneuver very easily through the many operation, configuration, and diagnostic screens.

Ease of Use

The Web browser configuration and console functions are an extensive subset of the functions provided by the command line interface. Because of its ease of use we recommend the use of the Web browser whenever possible.

Note: Functions that cannot be accessed via the Web browser interface are: the firmware functions (see 2.7, "MSS Server Firmware" on page 47) and some of the GWCON and Config functions (for example, PING).

2.5.1 Accessing the MSS Server Using the Web Interface

To access the MSS Server through its Web interface requires IP connectivity between the management workstation and the MSS Server. Any of the folllowing four mechanisms can be used:

- Locally, over the service port with a workstation that communicates using the serial line IP (SLIP).
- Remotely, over an external modem connection to the service port with a workstation using SLIP.
- Remotely, using the integrated Voice/Data/Fax or Data/Fax PCMCIA modem. This connection also runs SLIP.
- Over the ATM adapter, through a LAN emulation client or a Classical IP interface. This requires a MSS server that has been configured, has been activted, and is attached to an operational ATM network.

Any Web browser that supports HyperText Markup Language (HTML) tables and clickable images can be used, for example, IBM WebExplorer Version 1.03 or higher, Netscape Navigator Version 1.1N or higher, or Mosaic Version 2.1.1 or higher.

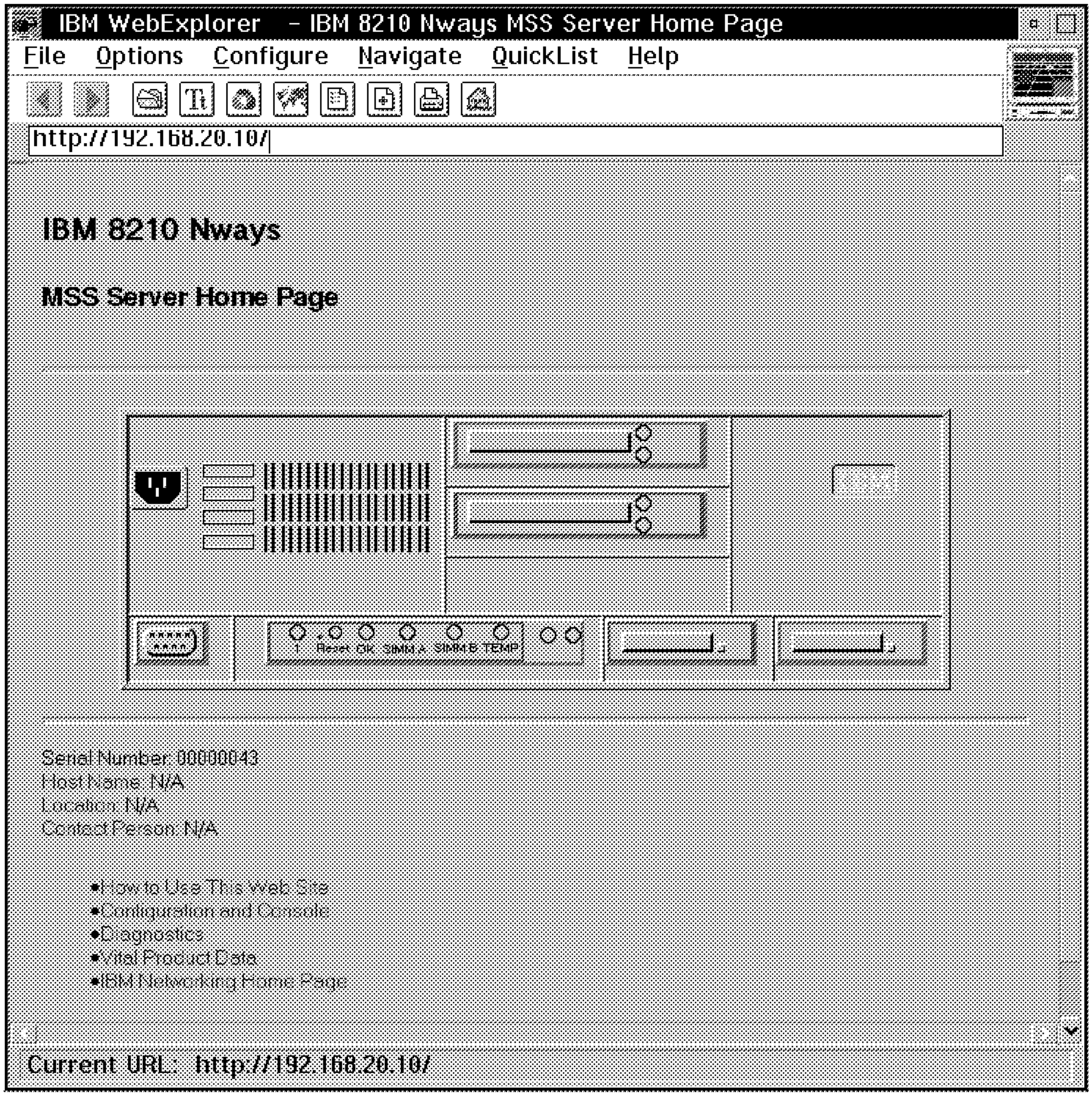
In order to access your MSS Server Home Page, you have to point your browser to URL http://<machine>/, where <machine> is either the name or one of the IP addresses configured on the MSS Server.

Using the Web interface via an in-band (ATM) connection requires that an IP address has been assigned to a LE or LIS client or that an LE client has been defined and the TCP/IP host services have been enabled. An alternative is to use out-of-band access via the SLIP interface. For details on how to establish a SLIP connection, see 2.1.2, “SLIP” on page 9.

Notes:

1. Many configuration options require you to enter data on multiple Web pages.
2. Two people should not perform configuration at the same time. They can interfere with one another such as deleting an interface that the other is working on.
3. You should disable the caching feature of the browser to avoid the browser to pull down a page out of memory instead of the latest information of the MSS Server.

The first form of the MSS Server Home Page is shown in Figure 17 on page 44.



http://192.168.20.10/

IBM 8210 Nways

MSS Server Home Page

Serial Number: 00000043
Host Name: N/A
Location: N/A
Contact Person: N/A

- How to Use This Web Site
- Configuration and Console
- Diagnostics
- Vital Product Data
- IBM Networking Home Page

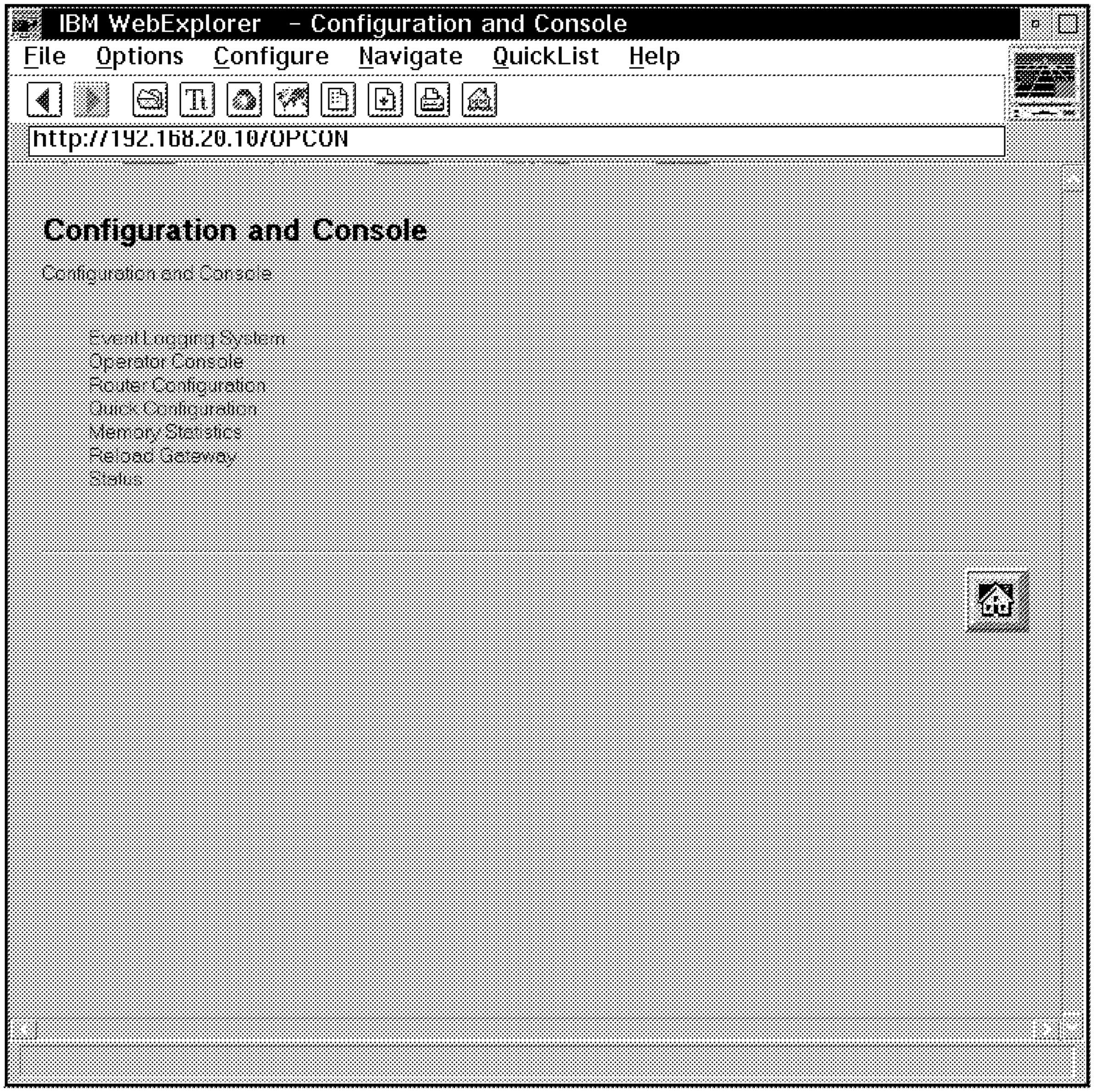
Current URL: http://192.168.20.10/

Figure 17. The MSS Server Home Page

This home page provides a graphical display showing the status of the MSS Server. It indicates the current network interfaces installed, the status of each port, the state of each LED and an indication of the devices that are installed in the two PCMCIA slots.

The page is refreshed every 80 seconds (your browser must support dynamic refresh). You can request immediate status in a separate, more detailed Web page, by simply clicking on the item. Hostname, location and contact person can be changed after clicking on it. The changes are immediately effective, however, to see them you to make sure your Web browser reloads the homepage.

Click on **How to use this Web Site** for instructions about using this site. Click on **Configuration and Console** to bring up the menu shown in Figure 18 on page 45.



http://192.168.20.10/OPCON

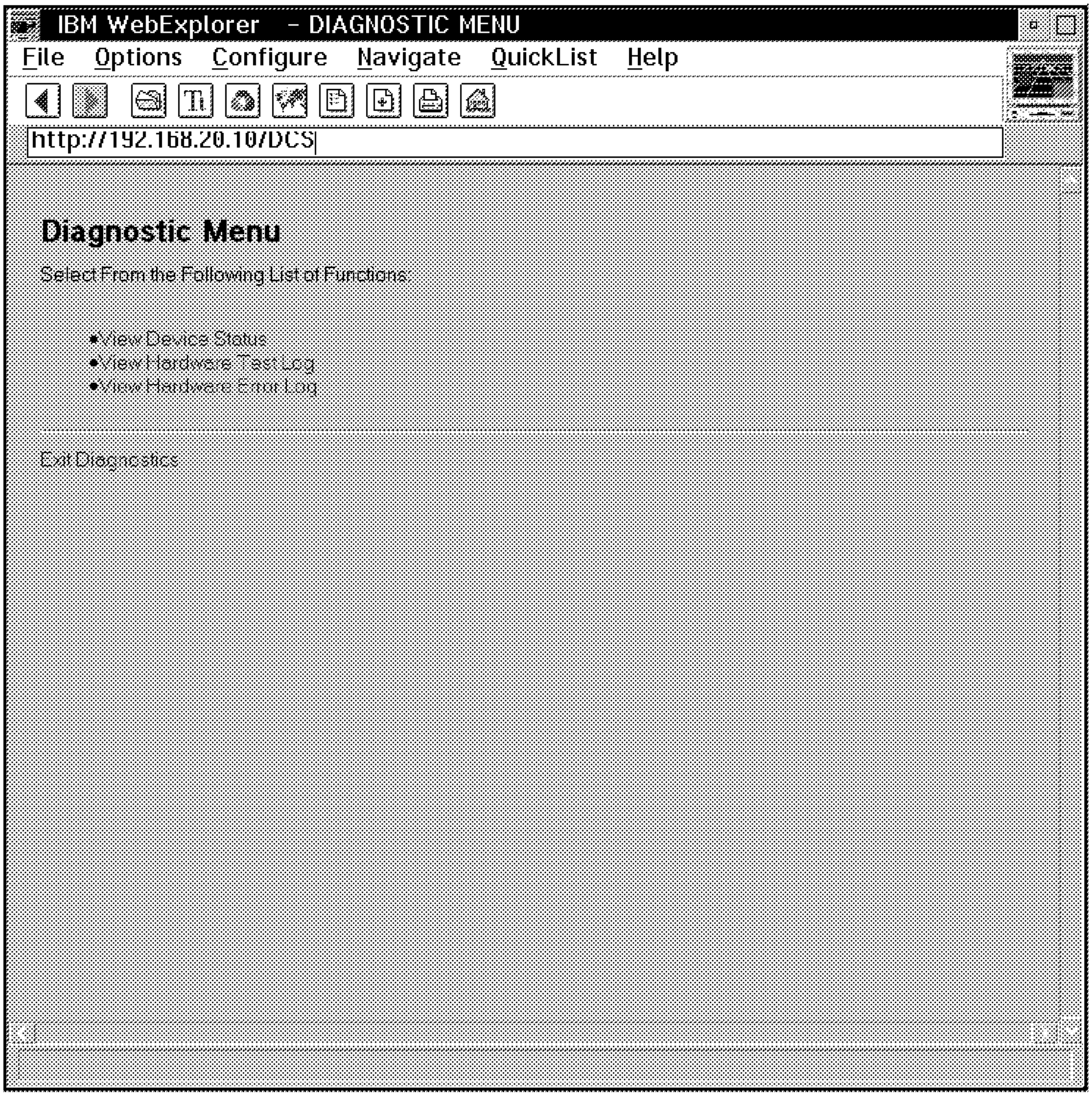
Configuration and Console

Configuration and Console

Event Logging System
Operator Console
Router Configuration
Quick Configuration
Memory Statistics
Reload Gateway
Status

Figure 18. The Configuration and Console Window

Click on **Diagnostics** to bring up the menu shown in Figure 19 on page 46.



http://192.168.20.10/DCS

Diagnostic Menu

Select From the Following List of Functions:

- View Device Status
- View Hardware Test Log
- View Hardware Error Log

Exit Diagnostics

Figure 19. The Diagnostics Window

Click on **Vital Product Data** for information about hardware and software.

Click on **IBM Networking Home Page** to navigate to this page (you must be connected to the Internet to use).

Important

Changes to the configuration that are made using the Web browser are written directly to the static random access memory (SRAM). The changes will take effect after a reload of the MSS Server. So, you can make unintentional changes that can create havoc after the next reload. It is important to check if you configured the MSS Server properly. Verify the settings for any parameters you have configured before submitting them.

2.6 Resetting the MSS Server

You can reset the MSS Server using one of these methods:

- Pressing the hardware reset button
- Typing `reload` at the `Config (only)>` prompt
- Typing `reload` at the OPCON prompt (*)

Attention

A reset interrupts the function of the MSS Server for up to 90 seconds. Be sure that the network is prepared for the interruption.

2.7 MSS Server Firmware

The MSS Server contains firmware and operational code. The operational code is required to run the actual 8210 functions, while the firmware is used for hardware and software management. The operational code runs on top of the firmware. Even if the operational code is not operational, the firmware will be active.

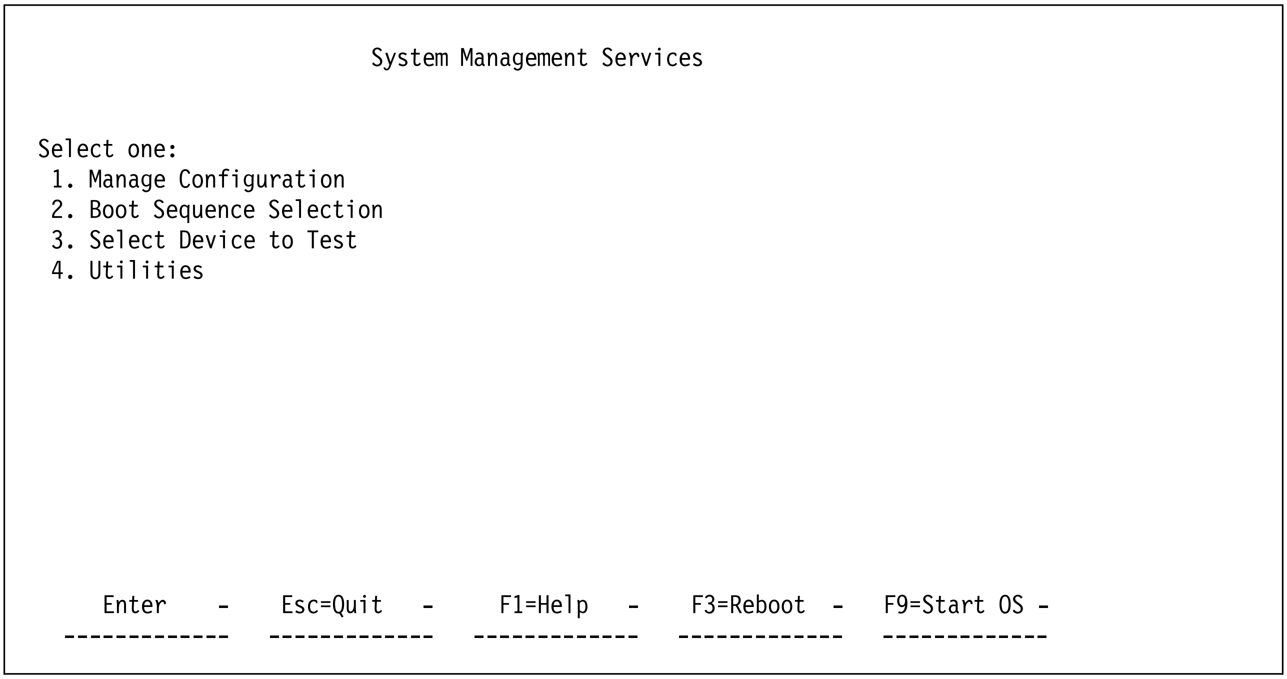
One of the functions is to perform hardware checking after a power-on, and decide which version of the operational code will be loaded. It also allows you to change some of the hardware related parameters, and manage the operational code and your configuration files.

In order to obtain operator access to the firmware, connect to the MSS Server using its service port, as discussed in 2.1, “Methods of Connecting” on page 7.

TTY Only

The firmware interface is only available via a TTY connection.

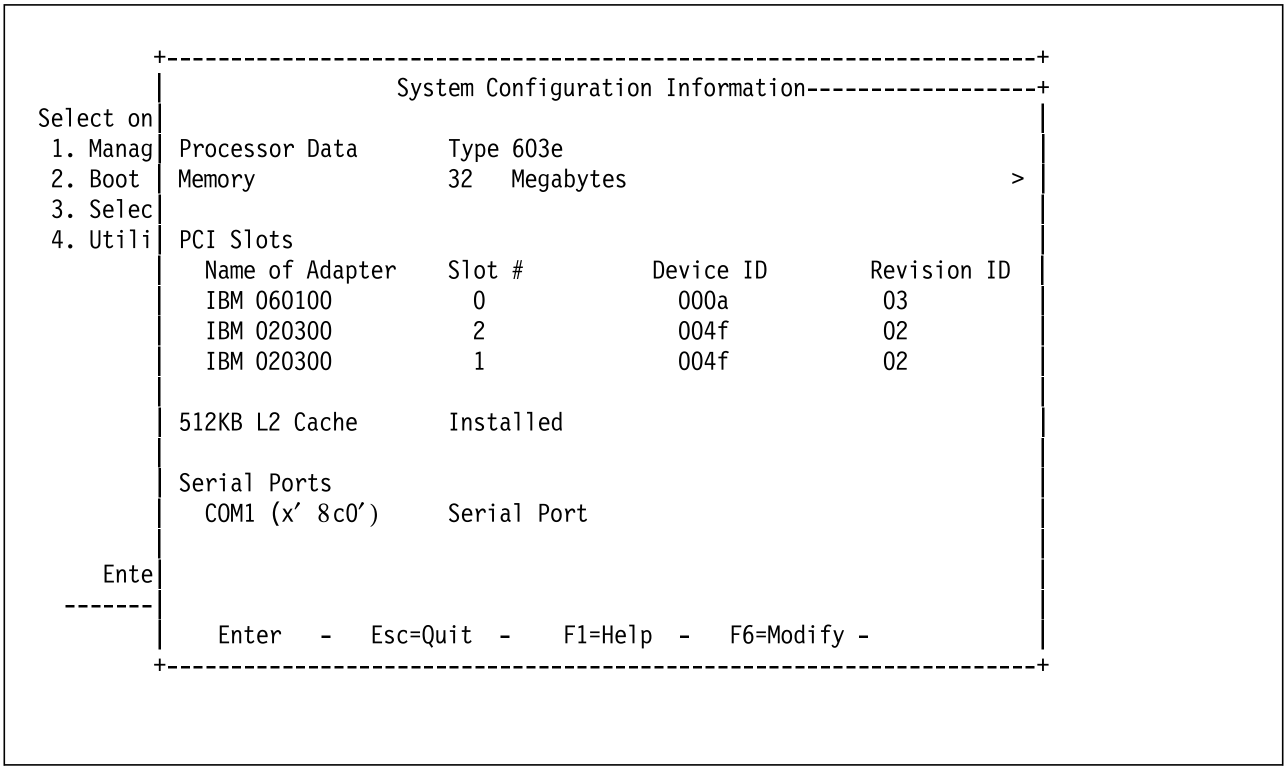
Power on, or reboot the machine and press and hold Ctrl+C during the boot sequence. At the end of the restart procedure you will be prompted for the supervisory password. Enter `mss` and the firmware menu depicted in Figure 20 on page 48 will appear.



```
                        System Management Services


  Select one:
   1. Manage Configuration
   2. Boot Sequence Selection
   3. Select Device to Test
   4. Utilities











       Enter    -    Esc=Quit   -     F1=Help   -    F3=Reboot  -   F9=Start OS -
    -------------   -------------   -------------   -------------   -------------
```

Figure 20. Firmware Menu

If you select the Manage Configuration option, the menu depicted in Figure 21 will appear.

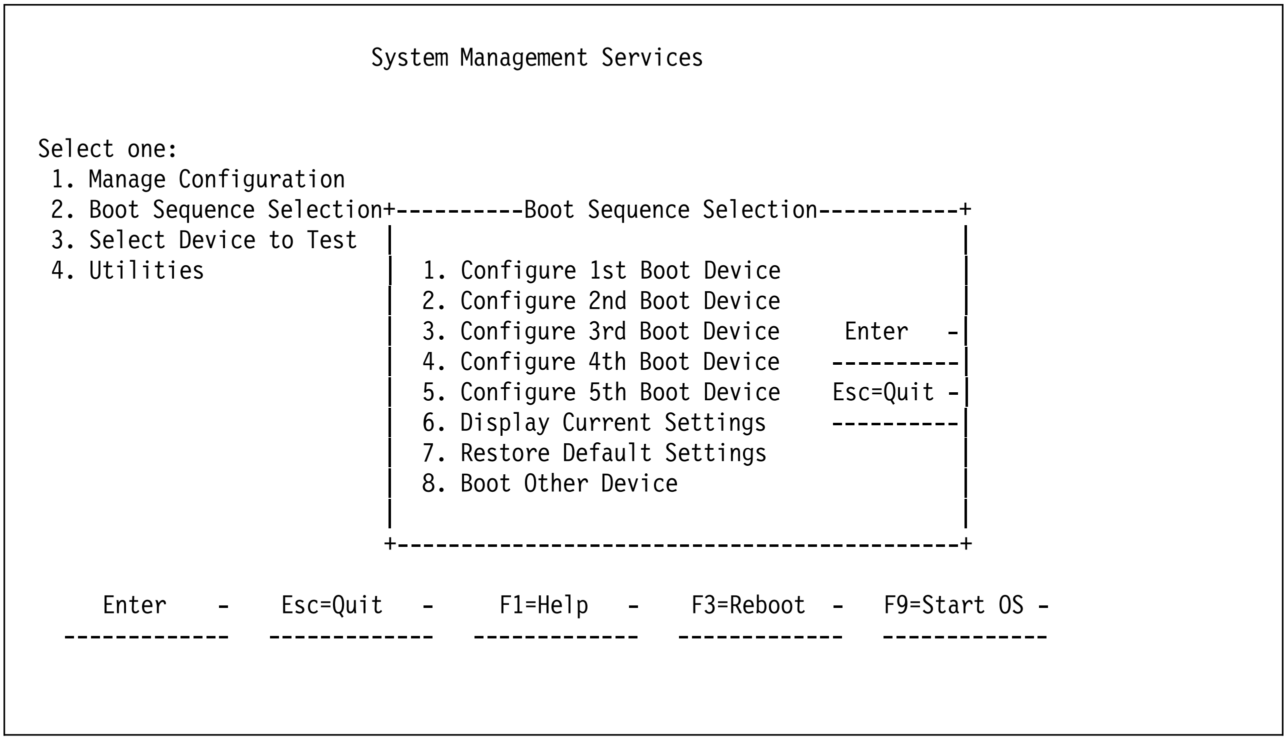


```
            +---------------------------------------------------------------------+
            |                   System Configuration Information------------------+
  Select on|                                                                      |
   1. Manag| Processor Data      Type 603e                                         |
   2. Boot | Memory              32   Megabytes                                  > |
   3. Selec|                                                                      |
   4. Utili| PCI Slots                                                            |
           |   Name of Adapter    Slot #         Device ID       Revision ID      |
           |   IBM 060100           0              000a            03             |
           |   IBM 020300           2              004f            02             |
           |   IBM 020300           1              004f            02             |
           |                                                                      |
           | 512KB L2 Cache      Installed                                        |
           |                                                                      |
           | Serial Ports                                                         |
           |   COM1 (x' 8c0')     Serial Port                                     |
           |                                                                      |
       Ente|                                                                      |
    -------|                                                                      |
           |    Enter   -   Esc=Quit  -    F1=Help  -   F6=Modify -               |
            +---------------------------------------------------------------------+
```

Figure 21. Manage Configuration Menu

You can, for example, use this menu to change the parameters of the serial port.

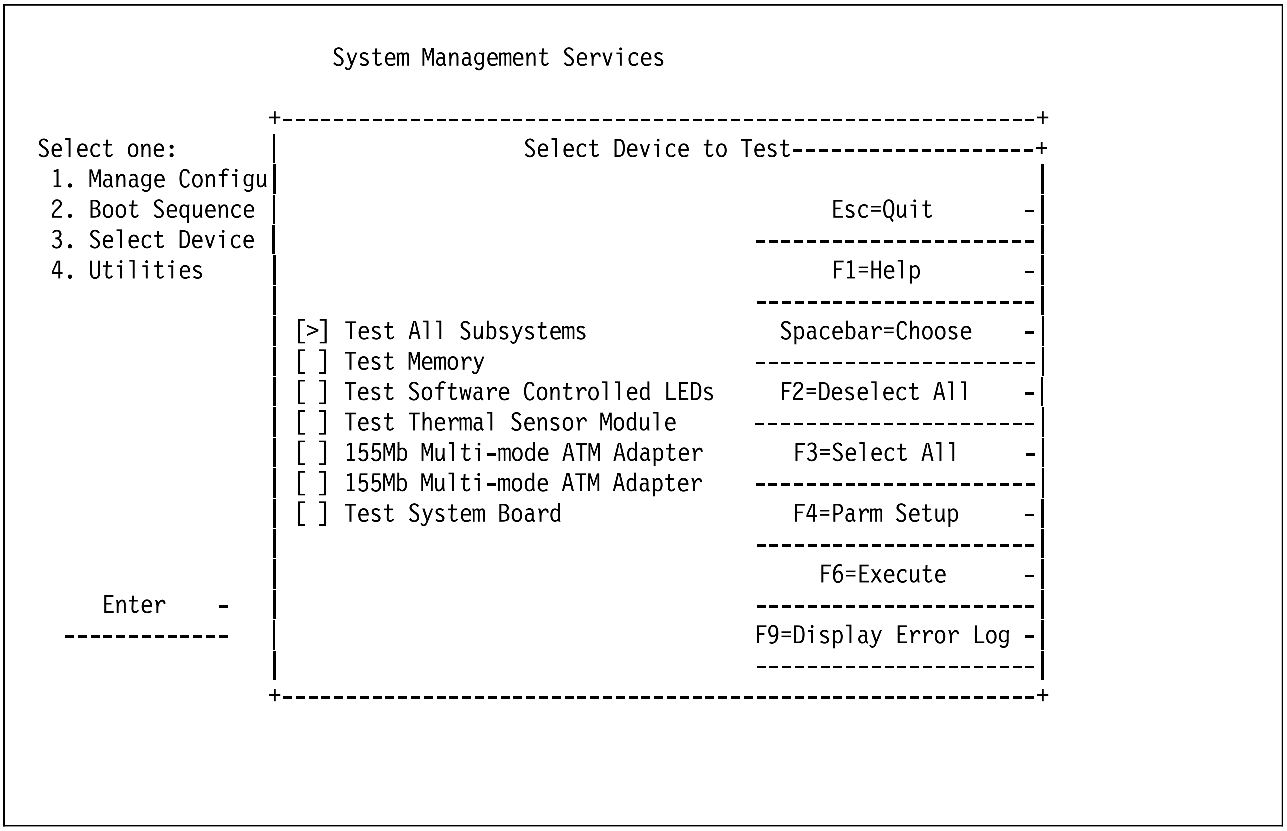
If you select the Boot Sequence Selection option from the main firmware management menu, you can specify if the 8210 should reboot using its operational code and configuration file stored in flash memory, or use the files stored on the (PCMCIA) hard disk. Figure 22 on page 49 shows the configuration options.



```
                           System Management Services


 Select one:
  1. Manage Configuration
  2. Boot Sequence Selection+----------Boot Sequence Selection-----------+
  3. Select Device to Test  |                                            |
  4. Utilities              |  1. Configure 1st Boot Device              |
                            |  2. Configure 2nd Boot Device              |
                            |  3. Configure 3rd Boot Device     Enter   -|
                            |  4. Configure 4th Boot Device    ----------|
                            |  5. Configure 5th Boot Device    Esc=Quit -|
                            |  6. Display Current Settings     ----------|
                            |  7. Restore Default Settings               |
                            |  8. Boot Other Device                      |
                            |                                            |
                            +--------------------------------------------+

      Enter    -    Esc=Quit   -     F1=Help   -    F3=Reboot  -   F9=Start OS -
   -------------   -------------   -------------   -------------   ------------
```

Figure 22. Boot Sequence Selection Menu

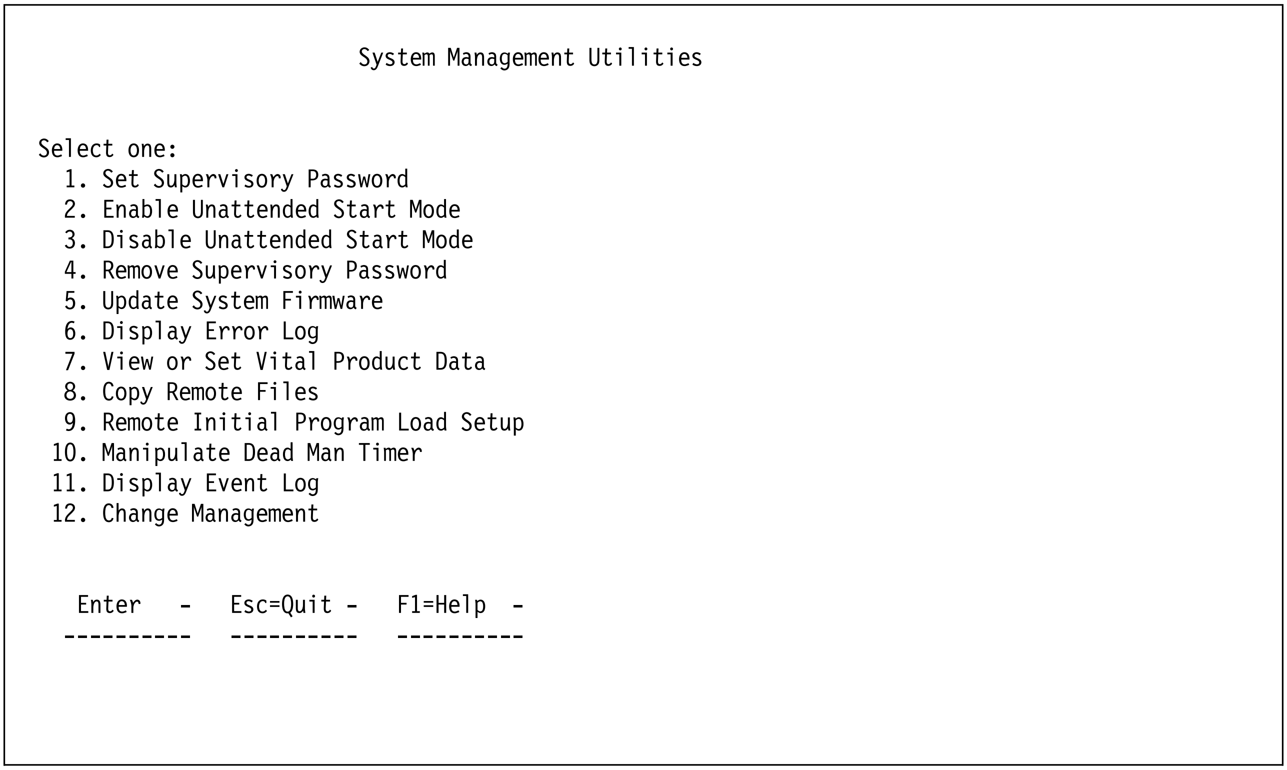
If you choose the Select Device to Test option from the main firmware management menu, Figure 23 on page 50 appears. This menu enables you to test specific hardware components of your 8210.



```
                        System Management Services

                    +------------------------------------------------------------+
 Select one:        |                  Select Device to Test-------------------+
  1. Manage Configu|                                                           |
  2. Boot Sequence |                                          Esc=Quit      -|
  3. Select Device |                                    ----------------------|
  4. Utilities     |                                          F1=Help       -|
                   |                                    ----------------------|
                   | [>] Test All Subsystems              Spacebar=Choose    -|
                   | [ ] Test Memory                      ----------------------|
                   | [ ] Test Software Controlled LEDs    F2=Deselect All    -|
                   | [ ] Test Thermal Sensor Module       ----------------------|
                   | [ ] 155Mb Multi-mode ATM Adapter      F3=Select All     -|
                   | [ ] 155Mb Multi-mode ATM Adapter     ----------------------|
                   | [ ] Test System Board                 F4=Parm Setup     -|
                   |                                      ----------------------|
                   |                                       F6=Execute        -|
     Enter    -    |                                      ----------------------|
   -------------   |                                    F9=Display Error Log -|
                   |                                      ----------------------|
                    +------------------------------------------------------------+
```

Figure 23. Select Device to Test Menu

Finally, by selecting Utilities in the main firmware management menu, Figure 24 appears.



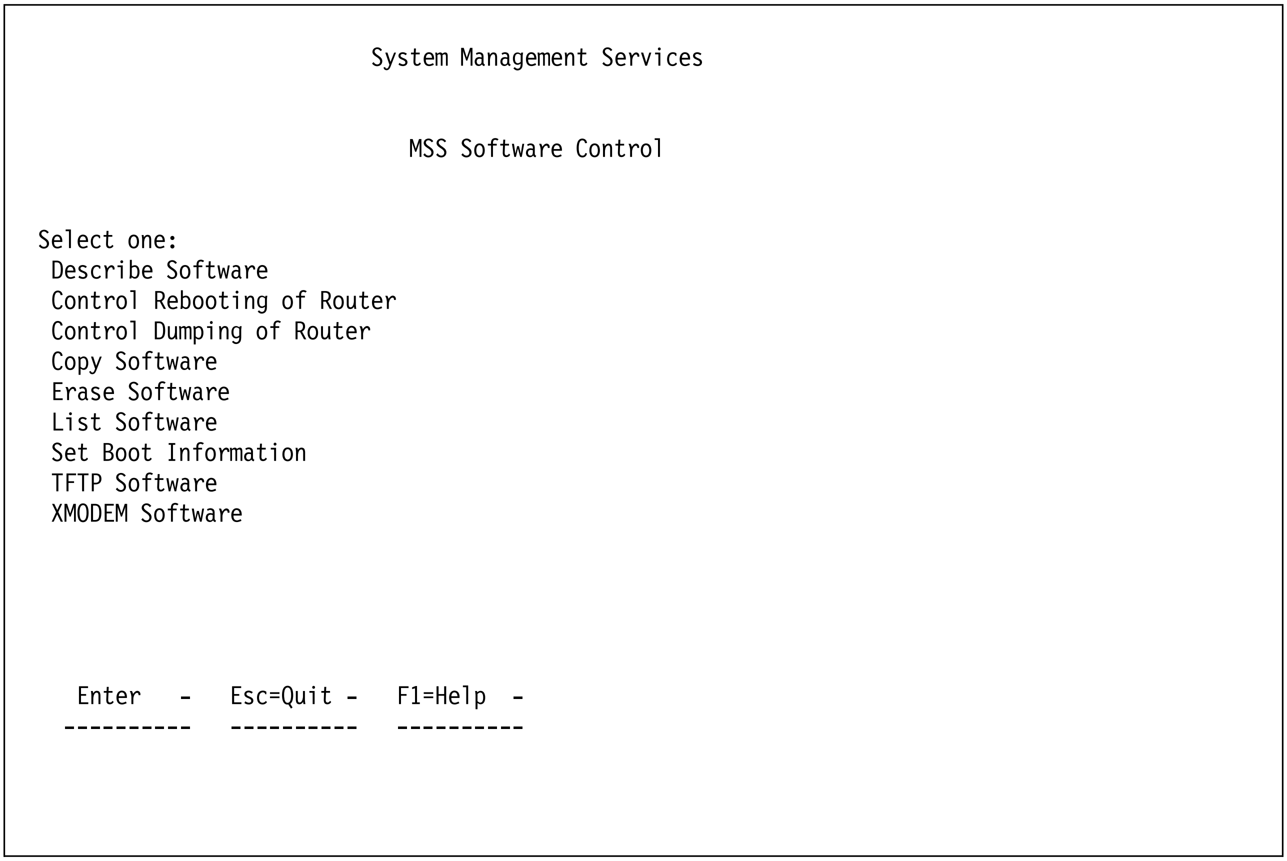
```
                          System Management Utilities


 Select one:
   1. Set Supervisory Password
   2. Enable Unattended Start Mode
   3. Disable Unattended Start Mode
   4. Remove Supervisory Password
   5. Update System Firmware
   6. Display Error Log
   7. View or Set Vital Product Data
   8. Copy Remote Files
   9. Remote Initial Program Load Setup
  10. Manipulate Dead Man Timer
  11. Display Event Log
  12. Change Management


    Enter   -   Esc=Quit -   F1=Help  -
   ----------   ----------   ----------
```

Figure 24. Utilities Menu

Here you can change the password, enable/disable unattended start mode, display information about the system as well as error and event logs among other options.

An important function is the management of the MSS operational code and configuration files, and deciding the order in which these files are selected during an 8210 restart.

These management functions, depicted in Figure 25, become available after selecting Change Management.



```
                      System Management Services


                         MSS Software Control


 Select one:
  Describe Software
  Control Rebooting of Router
  Control Dumping of Router
  Copy Software
  Erase Software
  List Software
  Set Boot Information
  TFTP Software
  XMODEM Software





    Enter   -   Esc=Quit -   F1=Help  -
   ----------   ----------   ----------
```

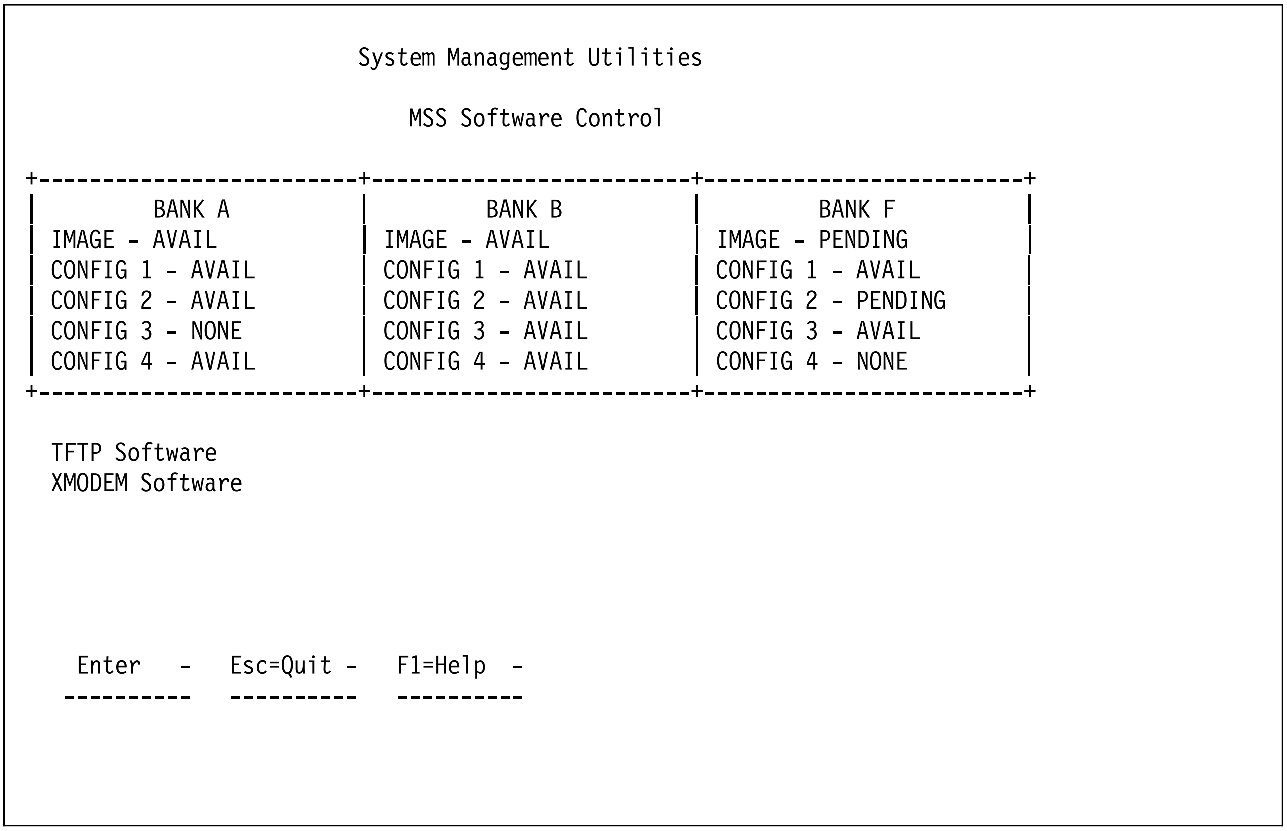
Figure 25. Change Management Menu

In this menu you can determine which operational code and which configuration file will be loaded during the next restart of the 8210. The List Software option in Figure 25 enables you to display the status of operational code and configuration files.

As depicted in Figure 26 on page 52, the MSS Server identifies three separate software banks. Bank A and B are maintained on the PCMCIA hard disk, while bank F is stored in the flash memory. On each of the banks you can store a copy of the operational code (IMAGE) and up to four configuration (CONFIG) files. Options exist to retrieve new operational code or configuration files, using TFTP or XMODEM.

The Set Boot Information option in Figure 25 enables you to indicate from which bank the 8210 will retrieve its operational code and configuration file. A backup bank can be defined which will be tried if loading of the operational code or configuration file is not succesful.

Note: The operational code that will be loaded must use one of the four configuration files present on the same bank.



```
                         System Management Utilities

                             MSS Software Control

+-------------------------+-------------------------+-------------------------+
|         BANK A          |         BANK B          |         BANK F          |
| IMAGE - AVAIL           | IMAGE - AVAIL           | IMAGE - PENDING         |
| CONFIG 1 - AVAIL        | CONFIG 1 - AVAIL        | CONFIG 1 - AVAIL        |
| CONFIG 2 - AVAIL        | CONFIG 2 - AVAIL        | CONFIG 2 - PENDING      |
| CONFIG 3 - NONE         | CONFIG 3 - AVAIL        | CONFIG 3 - AVAIL        |
| CONFIG 4 - AVAIL        | CONFIG 4 - AVAIL        | CONFIG 4 - NONE         |
+-------------------------+-------------------------+-------------------------+

  TFTP Software
  XMODEM Software





    Enter   -   Esc=Quit -   F1=Help  -
   ----------   ----------   ----------
```

Figure 26. List of the Software Status

Besides the operational code (IMAGE) and configuration files (CONFIG) of the bank, you can see its status. The meaning of this field is:

ACTIVE The file is currently loaded in the active memory and running on the MSS Server. The Active status will not be displayed using the firmware change management facilities.

Note: Using the `Boot Config>List Software` command results in a similar screen. The loaded configuration file and operational code will be indicated as Active.

AVAIL The file is good and can be made active.

BROKEN The file is damaged or was not completely loaded into the MSS Server.

PENDING The file will be loaded and become active after the next reset.

LOCAL The file will be loaded and become active after the next reset and the currently active file will become pending. Local is a status that makes a file active only for one reset.

NONE There is no file stored.

For further information about firmware, see the *Multiprotocol Switched Services (MSS) Server Service Manual*.

Chapter 3. MSS Server and ATM Ports

This section gives an overview of how to configure an ATM port on the IBM 8210 Nways MSS Server. Before detailing the command line configurator and the configuration program, we discuss the configuration parameters that can be defined.

3.1 ATM Port Parameters

When configuring an ATM port on the IBM 8210 Nways MSS Server, the following parameters should be carefully considered:

Note: We hereby refer to the definition keywords used by the command line configurator. The corresponding parameter field used by the configurator program can be easily derived.

network The *network* parameter specifies which ATM port is being configured. Specify either 0 or 1, to refer to ATM port 1 or 2, respectively.

Note: The 8260 MSS Server blade provides a single ATM attachment.

max-data-rate The *max-data-rate* value defines the speed of the physical ATM interface. The only value supported is 155 (Mbps).

Note: In the Configuration Program this parameter is referred to as *maximum VCC data rate*.

max-frame The *max-frame* parameter defines the maximum AAL5 CPCS PDU payload size. Any of the frames received by, or sent from, the 8210 must not exceed this value. Make sure that the maximum frame sizes defined on your ELANs and LISs do not exceed this value. The default value is 9234.

max-calls The *max-calls* parameter defines an upper boundary to the number of simultaneously active, point-to-point (PtP) and point-to-multipoint (PtMP), VCCs on the 8210. This parameter includes the VCCs established by the 8210, and the VCCs started by remote clients. The default value is 1024.

For a discussion of which VCCs are used by the 8210, see 3.5, “VCC Considerations” on page 61.

max-callers The *max-callers* parameter sets an upper boundary to the number of LE clients, LES/BUS, LECS, IPX (RFC 1483) clients, LIS clients, and ARP servers that can be simultaneously active on the 8210. The default value is 209.

max-config-selectors The *max-config-selectors* specifies the maximum number of selectors bytes (SELs) that can be specified per end system identifier (ESI). This number includes the manually assigned ESIs, and the ESIs generated by the configurator. The default value is 200.

Note: The remaining ESI numbers (out of a total of 256 per ESI) are used by the 8210 to assign ESIs at run-time.

max-mp parties The *max-mp parties* defines the maximum number of leaves (or parties) that any of the 8210 components, most notably the LES and BUS instances, accept on their point-to-multipoint connections. Note that this number may limit the number of LE clients that can join a single ELAN. The default value is 512.

trace The *trace* parameter, if on, allows tracing on specific VPI/VCI ranges. By default tracing is disabled.

uni-version The *uni-version* parameter specifies the UNI version that the 8210 uses on its ATM connection. This value can be 3.0, 3.1 or AUTO. AUTO will let the 8210 discover whether the adjacent switch supports UNI 3.0 or 3.1 and configures accordingly. Same values have to be defined on both sides of the ATM link. If UNI=AUTO has been configured at both ends, the latest UNI version supported (that is, UNI Version 3.1 for the 8210 and 8285/8260) will be selected. By default UNI Version 3.0 is used.

For more details, see the discussion in 3.1.1, "User-to-Network Interface (UNI)."

When configuring your LECS, LES/BUS and LE clients on the MSS Server, you have to specify which end system identifier (ESI) is used to construct the 20-byte ATM address (see 4.3, "ATM Addresses" on page 69) used to refer to it. Hereby you have the option to use the burned-in ESI of the ATM port it is associated with, or you can use a locally administered ESI. The use of locally administered ESIs has the advantage that the ATM addresses used to address components such as LECS, LES/BUS and LE clients are less likely to change, are easier to define on devices such as ATM switches and remote LE clients, and, because they are easier to recognize, simplify problem determination.

An ESI is six bytes long. It is associated with a specific ATM port and must be unique. In general, it is sufficient to define one locally administered ESI per ATM port.

3.1.1 User-to-Network Interface (UNI)

The IBM 8210 Nways MSS Server supports UNI V3.0 or UNI V3.1 on its ATM ports. When configuring an ATM port, you have to either hard-code one of these values or specify UNI auto-detection. In the last case the 8210 will conform to the UNI version configured on the adjacent ATM switch. When both ends have configured UNI=AUTO, they will learn about the highest UNI version on the other end using the ILMI GET UNI request. If the query fails, the MSS Server defaults to UNI Version 3.0.

IBM 8285/8260

Since the IBM 8285 and 8260 support both UNI versions, the IBM 8285/8260 will return the highest UNI version, that is, UNI Version 3.1.

Table 6 on page 55 depicts the valid combinations of UNI versions that can be configured on the 8210 and the adjacent 8260, or 8285, ATM switch.

Table 6. Valid UNI Combinations

8210	**8260/8285**		
	UNI=V3.0	**UNI=V3.1**	**UNI=AUTO**
UNI=V3.0	UNI=V3.0	invalid	UNI=V3.0
UNI=V3.1	invalid	UNI=V3.1	UNI=V3.1
UNI=AUTO	UNI=V3.0	UNI=V3.1	UNI=V3.1

In addition to the UNI version used on the IBM 8210 Nways MSS Server and its adjacent ATM switch, care should be taken with the UNI version used on remote ATM equipment.

In general, it is recommended that you use the same UNI version on all your ATM-attached equipment. Use UNI Version 3.1 whenever possible. If for any reason this is not possible, make sure that you connect your equipment to ATM switches, such as the IBM 8260 Multiprotocol Intelligent Switching Hub and IBM 8285 Nways ATM Workgroup Switch, that provide translation of UNI connection requests.

3.2 Using the Command Line Configurator

Figure 27 on page 56 depicts how to list the ATM port parameters for network 0 (that is, port 1) and assign a locally administered ESI to it. Note, port 1 is present by default and need not be added.

```
*talk 6
Config>network 0
ATM user configuration
ATM Config>interface 0
ATM interface configuration
ATM Interface Config>set ?
MAX-DATA-RATE
MAX-FRAME
MAX-CONFIG-SELECTORS
MAX-CALLS
MAX-CALLERS
MAX-MP-PARTIES
TRACE
UNI-VERSION
ATM Interface Config>list configuration
                         ATM Configuration
  Interface (net) number =    0
  Maximum VCC data rate Mbps   =     155
  Maximum frame size     = 9234
  Maximum number of callers =  209
  Maximum number of calls = 1024
  Maximum number of parties to a multipoint call =  512
  Maximum number of Selectors that can be configured  =  200
  UNI Version = UNI 3.1
  Packet trace = OFF
ATM Interface Config>
```

Define ESI, exit and reload.

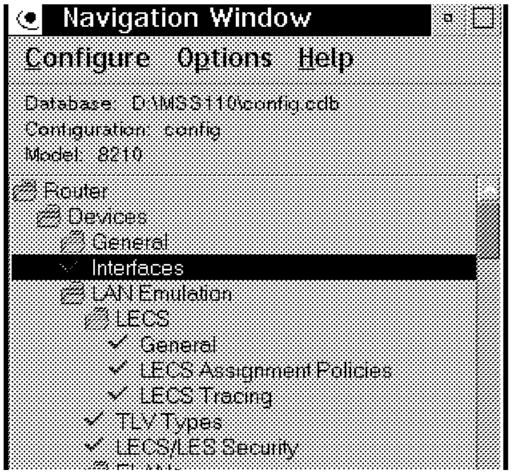
```
ATM Interface Config>add esi 40.00.00.82.10.00
ATM Interface Config>exit
ATM Config>exit
Config>[CTRL-P]
*reload
```

Figure 27. ATM Port Configuration - Using Line Commands

The figure shows how you can add an ESI using the `add esi` command. An overview of the configurable parameters is obtained with the `set ?` command (the parameters listed are detailed in 3.1, “ATM Port Parameters” on page 53). Finally, using the `list configuration` command, all configurable parameters and their values are listed.

3.3 Using the Configuration Program

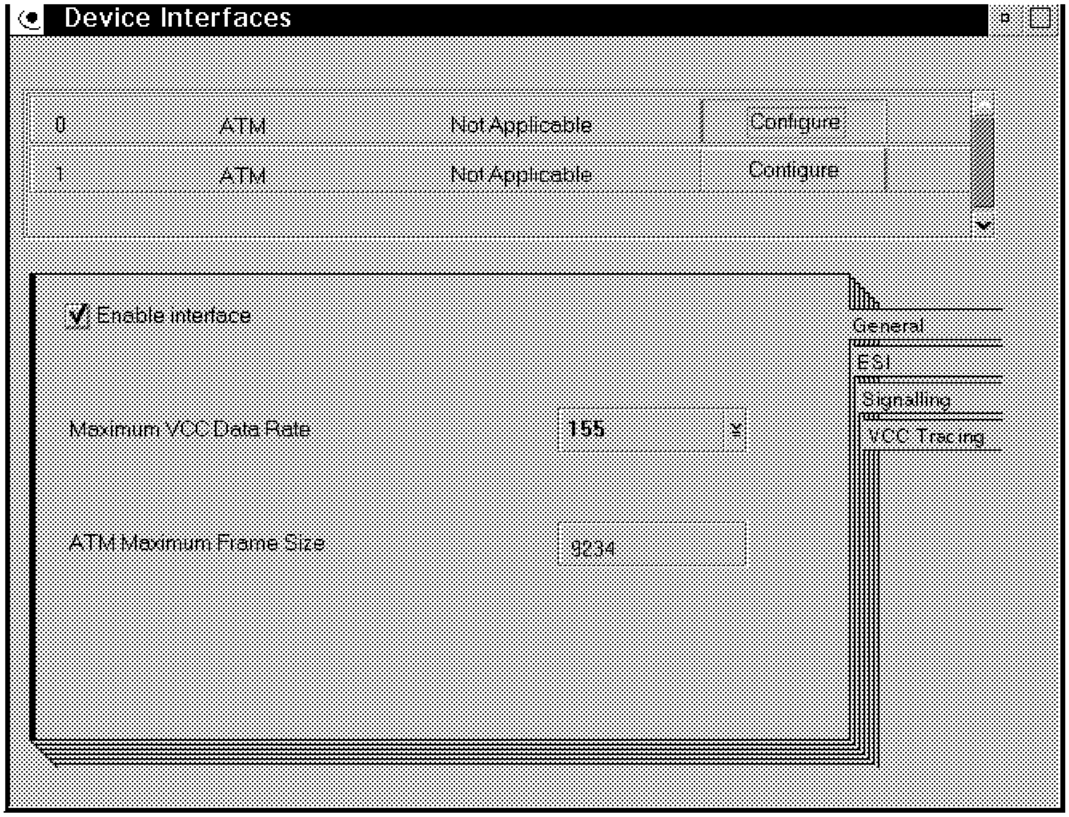
Figure 28 on page 57 depicts the Navigation Window of the configuration program. Selecting Interfaces enables you to specify configuration details for the ATM ports.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 28. Interface Navigation

After selecting the ATM interface you want to configure and clicking on General, Figure 29 appears.



0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

☑ Enable interface

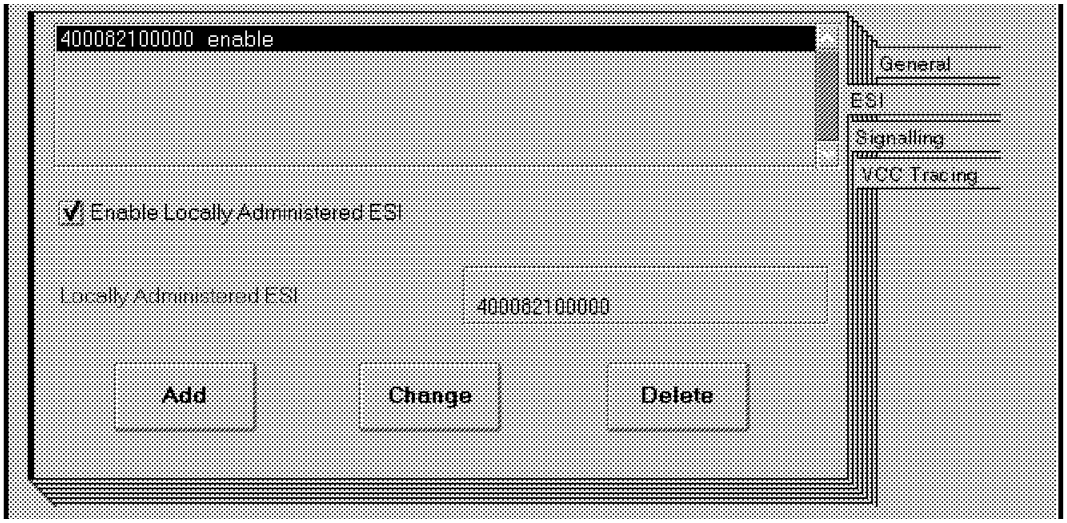
Maximum VCC Data Rate 155

ATM Maximum Frame Size 9234

Figure 29. Enabling ATM Port

Figure 30 on page 58 depicts the three submenus that become available after clicking on ESI, Signaling, and VCC Tracing, respectively, within Figure 29. See 3.1, “ATM Port Parameters” on page 53 for a description of the parameters that can be entered.

ESI:

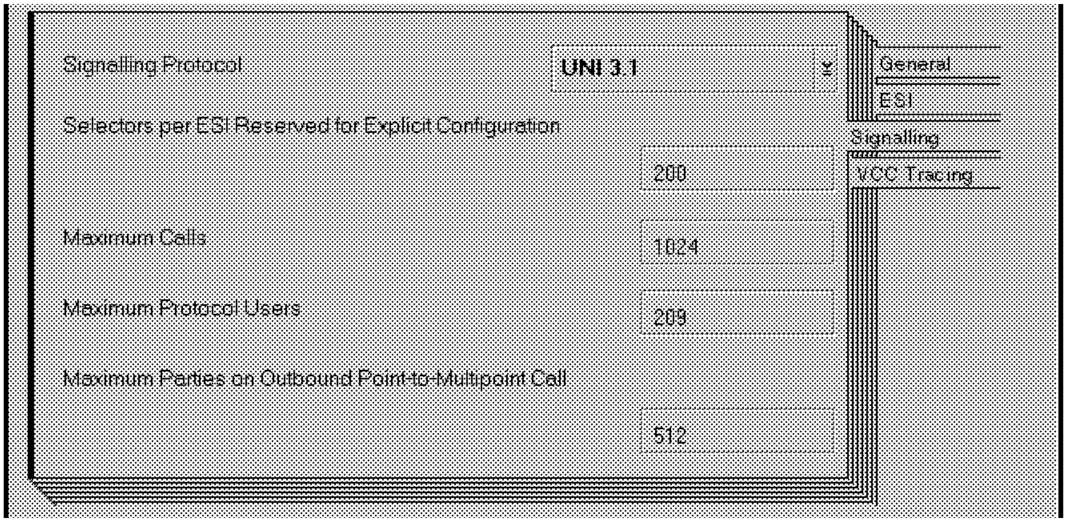


400082100000 enable

☑ Enable Locally Administered ESI

Locally Administered ESI: 400082100000

Signalling:



Signalling Protocol: UNI 3.1

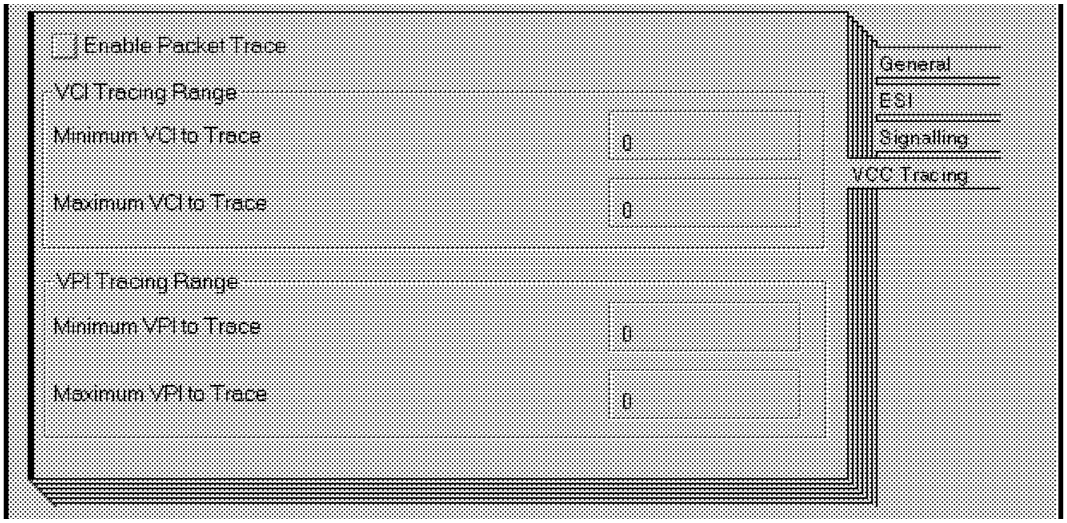
Selectors per ESI Reserved for Explicit Configuration: 200

Maximum Calls: 1024

Maximum Protocol Users: 209

Maximum Parties on Outbound Point-to-Multipoint Call: 512

VCC Tracing:



□ Enable Packet Trace

VCI Tracing Range

Minimum VCI to Trace: 0

Maximum VCI to Trace: 0

VPI Tracing Range

Minimum VPI to Trace: 0

Maximum VPI to Trace: 0

Figure 30 (Part 3 of 3). ATM Port Configuration - Using the Configuration Program

3.4 Configuring the Adjacent ATM Switch

To enable the IBM 8210 Nways MSS Server to connect to the ATM network, it is important that on the adjacent ATM switch an appropriate UNI interface is specified.

Note: Table 6 on page 55 depicts the valid UNI combinations defined on the IBM 8210 Nways MSS Server and the adjacent ATM switch.

Figure 31 shows the `set port` command to define an IBM 8210 Nways MSS Server UNI V3.1 attachment. In our example we have connected the 8210 to port 14.1 of the ATM switch. Make sure that in your configuration port 14.1 is replaced with the actual port. Note that the ILMI function has been enabled. We used the `show port` command to display the status of the line.

```
8260ATM1> set port 14.1 enable ilmi_forced_sig_3_1
Port set
8260ATM1> show port 14.1 verbose

     Type  Mode     Status
-------------------------------------------------------------------------------
14.01:UNI enabled  UP-OKAY

Signalling Version   : with ILMI, forced 3.1
Flow Control         : On
Frame format         : SDH STM-1
Connector            : SC DUPLEX
Media                : Multimode fiber
Port speed           : 155000 Kbps
Remote device is active
IX status            : IX OK
Scrambling mode      : frame and cell
Clock mode           : internal
```

Figure 31. ATM Interface Definition

When defining an ARP server and/or client on the IBM 8210 Nways MSS Server, its 20-byte ATM address is used to establish client-to-client or client-to-server connections. This address is composed of a concatenation of:

- 13-byte network identifier, learned from the adjacent ATM switch
- 6-byte end system identifier (ESI), defined on the 8210
- 1-byte selector (SEL), defined on the 8210

Figure 32 depicts how the 20-byte ATM address of the CPSW on the 8260/8285 ATM switch can be displayed using the show device command. The first 13 bytes of this address is the 13-byte network identifier used by all devices that attach to this switch.

```
8260ATM1> show device
8260 ATM Control Point and Switch Module
Name : 8260A
Location :
ITSO Raleigh, B678

For assistance contact :
Jaap de Goede

Manufacture id: VIME
Part Number: 58G9605 EC Level: C38846
Boot EEPROM version: v.1.2.0
Flash EEPROM version: v.2.0.4
Flash EEPROM backup version: v.2.0.4
Last Restart : 12:37:21 Fri 20 Sep 96 (Restart Count: 66)

A-CPSW
---------------------------------------------------------------------------
 ATM address: 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.82.60.A1.00
```

Figure 32. ATM Interface Definition

Note: The previous command may be useful when an ARP server or LIS client′s ATM address must be specified during the configuration of an LIS client on another device.

3.4.1 Verifying ESI Registration on the ATM Switch

Figure 33 depicts an easy method of verifying if the ATM attachment, port and ESI definitions were successful. It shows the show atm_esi command, which display the ESIs that the IBM 8210 Nways MSS Server has successfully registered.

```
8260ATM1> show atm_esi
Enter module: 14.
Enter port: 1
Port   ATM_ESI           Type
---------------------------------------------------------------------------
14.01 50.00.82.10.00.00 dynamic
8260ATM1>
```

Figure 33. Verifying ESI Registration

Note: In our configuration the 8210 connects to port 14.1 of the adjacent 8260 ATM switch. Replace port 14.1 with the actual ATM attachment when verifying your configuration.

3.5 VCC Considerations

The IBM 8210 Nways MSS Server uses point-to-point (PtP) and point-to-multipoint (PtMP) VCCs for the following functions:

1. LAN Emulation

   Identify hereby the VCCs used by the following components:

   - LAN emulation configuration server (LECS)

     – A PtP (*configuration direct*) VCC to/from each LE client

       Note: Most LE clients will drop their configuration direct VCC after they have completed their registration procedure.

     – Optionally, PtP (*security*) VCCs to remote 8210s

   - LAN emulation server (LES)

     – One or two (intelligent LES) PtMP (*control distribute*) VCC(s) to each LE client
     – A PtP (*control direct*) to/from each LE client
     – Optionally, one PtP (*security*) VCC to/from the LECS
     – Optionally, one PtP (*redundancy*) VCC to/from the partner LES

   - Broadcast and unknown server (BUS)

     – One or two (intelligent BUS) PtMP (*multicast forward*) VCC(s) to each LE client
     – A PtP (*multicast send*) to/from each LE client

   - LE clients

     – One PtP (*configuration direct*) to/from the LECS

       Note: The 8210 will drop the configuration direct VCC as soon as its client has joined an ELAN.

     – One PtP (*control direct*) to/from the LES
     – One PtP (*multicast send*) to/from the BUS
     – PtP (*data direct*) VCCs to/from other LE clients

       In addition, each LE client will be:
       - A party in the PtMP (*control distribute*) VCC from the LES
       - A party in the PtMP (*multicast forward*) VCC from the BUS

2. Classical IP

   Identify hereby the VCCs used by the following components:

   - ARP Server

     – One PtP VCC for each LIS client

   - LIS client

     – One PtP VCC to/from the ARP Server (only if client is using an ARP server and is no ARP server itself)
     – PtP data direct VCCs to/from other LIS clients

As an example, consider the VCC requirements for an IBM 8210 Nways MSS Server that is part of:

1. One emulated LAN with a local LECS, local LES/BUS, and 100 LE clients

   It is assumed that all LE clients are external, and all connect to the LECS. No intelligent LES, no intelligent BUS, no ELAN security and no LES/BUS redundancy are used.

   The VCC requirements for this ELAN are:

   LECS 100 *configuration direct* PtP VCCs

   LES 100 *control direct* PtP VCCs, one *control distribute* PtMP VCC

   BUS 100 *multicast send* PtP VCCs, one *multicast forward* PtMP VCC

   Note: For this ELAN the 8210 participates in 300 PtP VCCs and 2 PtMP VCCs. The 100 configuration direct VCCs are likely to be dropped after the LE clients have completed their registration procedure.

2. One emulated LAN with a local LECS, local LES/BUS, remote (backup) LES/BUS, and 100 LE clients

   It is assumed that all LE clients are external, and all connect to the LECS. Intelligent LES, intelligent BUS, ELAN security and LES/BUS redundancy are used.

   The VCC requirements for this ELAN are:

   LECS 100 *configuration direct* PtP VCCs, two *security* PtP VCCs

   LES 100 *control direct* PtP VCCs, two *control distribute* PtMP VCCs, one *security* PtP VCC, one *redundancy* PtP VCCs

   BUS 100 *multicast send* PtP VCCs, two *multicast forward* PtMP VCCs

   Note: For this ELAN the 8210 participates in 304 PtP VCCs, and 4 PtMP VCCs.

3. One emulated LAN with one LE client

   It is assumed that the LECS, and LES/BUS are external, and the LE client connects to 10 remote LE clients.

   The VCC requirements for this ELAN are:

   LE client 10 *data direct* PtP VCCs, one *configuration direct* PtP VCC, one *control direct* PtP VCC, and one *multicast send*) PtP VCC.

   Note: The configuration direct VCC will be dropped after the LE client completes its registration.

   In addition the LE clients is party in the *control distribute* and the *multicast forward* PtMP VCCs.

   Note: For this ELAN the 8210 participates in 12 PtP VCCs, and is party in 2 PtMP VCC.

4. One logical IP subnet (LIS) with a local ARP server

   It is assumed that this LIS contains 50 external LIS clients that are all connected to the ARP server.

   The VCC requirements for this LIS are:

   ARP server 50 *control data* PtP VCCs

   Note: For this LIS the 8210 participates in 50 PtP VCCs.

5. One logical IP subnet (LIS) with a local LIS client

   It is assumed that the LIS contains 10 external LIS clients that are all interconnected.

   The VCC requirements for this LIS are:

   LIS client 10 *data direct* PtP VCCs, one *control data* PtP VCC

   Note: For this LIS the 8210 participates in 11 PtP VCCs.

The VCC requirements grow rapidly when the number of ELANs and LISs to which the 8210 connects increases. Especially when performing ELAN and/or LIS server functions, the number of VCCs required can become considerable.

Because its networking resources are not unlimited, the number of local and remote clients that can be serviced by the 8210 is bound. When designing your network make sure that:

- You do not exceed the maximum VCCs supported on the adjacent switch

  See the discussion in 3.5.1, "Number of VCCs on 8260 ATM Switches."

- You do not exceed the storage and throughput capacity of the 8210

  BUS, IP routing, IPX routing, and bridging functions may cause considerable 8210 utilization in terms of storage and throughput.

  Note: For IBM 8210 Nways MSS Server capacity and performance details, see the *MSS*[1] *White Paper*.

- You do not exceed the maximum VCCs supported on the 8210

  The maximum number of VCCs per ATM port is 10,000. This high number is unlikely to limit your connectivity, however, be aware that the actual number of VCCs may be limited due to the values set during configuring your ATM port. Hereby the *max-calls* parameter provides an upper boundary to the number of PtP and PtMP VCCs, and *max-mp-parties* maximizes the number of parties within each PtMP VCC. Default values for these parameters are 1024 and 512, respectively.

  Note: For a discussion of these and other parameters, see 3.1, "ATM Port Parameters" on page 53.

3.5.1 Number of VCCs on 8260 ATM Switches

It has been discussed in the previous section that the 8210 functions may require a considerable number of VCCs. As pointed out, due to VCCs constraints, the ATM switch adjacent to the 8210 may become a limiting factor in the size and number of ELANs and logical IP subnets.

When attaching the 8210 to an IBM 8260 Multiprotocol Intelligent Switching Hub, the following VCC figures apply:

- The maximum number of PtMP connections per 8260 is 127.

  No limitations per module exist.

  Note: Be aware that one PtMP connection takes PtMP control blocks in all 8260s it traverses.

[1] The *MSS White Paper* is a technical publication that can be obtained from *http://www.raleigh.ibm.com/tr2/tr2over.html.* IBM employees can get a copy from MKTTOOLS as well.

- The maximum number of PtMP parties per 8260 is 3072.

  The 8260 has 6144 control blocks for party control blocks and requires 2 control blocks per PtMP connection. This limits the number of clients that can be added to PtMP VCCs to 3072.

- The maximum number of PtP connections per 8260 is 6144.

  The 8260 A-CPSW has 12,288 connection control blocks. Per PtP connection, the A-CPSW requires two control blocks.

- The maximum number of PtP connections per 8260 module is 4064.

  Each 8260 module has 4064 connection control blocks. Each PtP connection requires one control block in the module to which the endpoints of the VCC connect (note, if both connect to the same module, the VCC consumes two control blocks).

  Note: The mentioned number applies to the new FPGA picocode shipped with CPSW Version 2.3 and 2.4. The old number is 992.

Important

If you have a 2-port 8210, connect the ATM ports to different modules on the 8260. If possible connect to different 8260s.

3.5.2 Capacity Characteristics

Using these figures and the VCC requirements discussed in 3.5, “VCC Considerations” on page 61 allows us to calculate the maximum of server and client instances on the 8210, and the maximum number of clients within the ELANs and LISs serviced by the 8210.

Table 7 depicts the maximum numbers for the following situations:

1. Single port 8210 attached to 8260

   Note: The same numbers apply for the 8260 MSS Server blade

2. The ports of a 2-port 8210 attached to different modules on the same 8260
3. The ports of a 2-port 8210 attached to separate 8260s
4. An 8210 attached to an ATM switch without VCC limitations

   Note: The number of ATM ports is irrelevant, as the maximum connectivity of the 8210 will be reached.

Important

Be aware that the table depicts the maximum numbers for each individual function. When multiple functions are used, the maximum for each individual function decreases.

Table 7 (Page 1 of 2). Maximum Number of Server and Client Instances

Capacity Characteristic	Hardware Configuration			
	(1)	**(2)**	**(3)**	**(4)**
LECS Instances	1	1	1	1
LECS Policy Values	1500	1500	1500	1500

Table 7 (Page 2 of 2). Maximum Number of Server and Client Instances

Capacity Characteristic	**Hardware Configuration**			
	(1)	**(2)**	**(3)**	**(4)**
LES/BUS instances	31-63[1]	31-63[1]	31-63[1]	300
Served LE clients	496-1536[2]	992-1536[2]	992-3072[2]	5000
Served LE clients per ELAN	496-1536[2]	496-1536[2]	496-1536[2]	5000
Internal LE clients	63[3]	63[3]	126[3]	5000
LIS client	32[4]	64[4]	64[4]	32 per ATM port
Served LIS client	992-3000[2,5]	1984-3000[2,5]	1984-3000[2,5]	3000

Note:

1. The lowest number applies when using an intelligent LES and an intelligent BUS. Note that the maximum number of PtMP connections on the ATM switch decides the maximum value.
2. The lowest number applies when using less-recent CPSW code on the 8260 (see 3.5.1, “Number of VCCs on 8260 ATM Switches” on page 63). Note that the maximum number of PtP connections on the ATM switch decides the maximum value.
3. Number is limited by the maximum number of PtMP connections on the 8260.
4. The 8210 supports up to 32 LIS clients (or client/server) per ATM port.
5. The maximum number of LIS clients per 8210 is 3000.

MSS White Paper

Some of the figures mentioned in the previous table differ (slightly) from those mentioned in the *MSS*[1] *White Paper* due to different ways of rounding.

Chapter 4. ATM Forum-Compliant LAN Emulation

This chapter is intended to introduce the basic concepts of ATM Forum-compliant (FC) LAN emulation. Reading this section assists you in understanding and implementing emulated LAN networks and enables you to understand and appreciate the LAN emulation value-adds included in the MSS Server. Where relevant we have included the specific referance to the IBM 8210 Nways MSS Server. For a detailed description of the LE value-adds, and for information on how to configure the MSS Server, refer to Chapter 5, "MSS Server and LAN Emulation" on page 87.

4.1 LAN Emulation Benefits

Today′s networking applications are running primarily on Ethernet and token-ring networks that interface to LAN adapters via standard interfaces such as ODI and NDIS. ATM APIs (application programming interfaces) are under development which will allow applications to interface directly with the ATM layer and take advantage of all of ATM′s advanced features (such as quality of service). In the meantime, a service is required that will allow existing applications running on Ethernet and token-ring networks to take advantage of at least some of ATM′s benefits today, such as high-speed switched connections and scalability. This service is called LAN emulation.

LAN emulation allows ATM-attached stations, as well as token-ring and Ethernet stations, to communicate over an ATM network without any changes to existing applications. For directly attached ATM stations (via 25, 100 or 155 Mbps), a driver is installed at the Data Link Layer, which presents a standard token-ring or Ethernet interface to the upper layers, while at the same time, converting LAN frames to ATM cells to present to the ATM network. Endstations that do not have an ATM interface, but are on token-ring or Ethernet segments, can access the ATM network by means of a bridge or switch that has an ATM uplink. This type of device is known as a proxy as it does the frame-to-cell conversion and connection management on behalf of its LAN-attached endstations.

IBM 8210 Nways MSS Server provides a fully compliant implementation of LAN Emulation 1.0, as specified by the ATM Forum. In addition, IBM 8210 Nways MSS Server provides a number of value-add features that extend the range of LAN emulation. The LE value-add features are described in detail in Chapter 5, "MSS Server and LAN Emulation" on page 87.

4.2 Emulated LANs (ELANs)

There is much confusion in the industry over what is meant by the terms ELAN and VLAN, as these terms are often used interchangably. To avoid further confusion, we define these terms as they apply in this publication.

VLAN: A VLAN (virtual LAN) is a logical grouping of hosts, independant of physical location in the network, which defines what stations can communicate to each other. A VLAN can be based, for instance, on layer 2 MAC addresses, in which case, all members of the VLAN belong to the same broadcast domain. VLANs can also be assigned based on LAN switch ports.

ELAN: The term ELAN (emulated LAN) is a specific implementation of a virtual LAN, as it relates to LAN emulation in ATM networks. An ELAN consists of one or more LAN emulation clients (LECs) that share the same LAN emulation server and broadcast and unknown server (LES/BUS). Broadcasts by any member of the ELAN are contained within the boundaries of that ELAN, and ELAN membership can be assigned based on configurable policies.

The MSS Server supports the creation of multiple ELANs, each with their own instance of LES/BUS. In addition, it also supports a single instance of LECS (LAN emulation configuration server). The LES, BUS and LECS are collectively known as the LE Service components. The LE clients may obtain the address of their LES/BUS from a LAN emulation configuration server (LECS) or, alternatively, may be preconfigured with their LES/BUS address. It is preferable to allow the LECS to assign the LES/BUS address to the LE client, as the LECS can act as the central administration point for the creation of ELANs based on policies. For instance, an ELAN based on an ELAN_NAME policy could assign all LE clients with the name ACCOUNTS to one ELAN, and all the LE clients with the name ENGINEERING to another ELAN. For ATM-attached endstations, this would allow the physical re-location of the endstation without requiring reconfiguration of that endstation, that is, an ENGINEERING workstation would still belong to the ENGINEERING ELAN, regardless of where it is located.

Note: For token-ring or Ethernet attached endstations, care must be taken when relocating endstations to ensure that the switch or bridge to which they are being moved has an LE client on the same ELAN.

Figure 34 shows a physical and logical view of a simple LAN emulation network consisting of two ELANs.

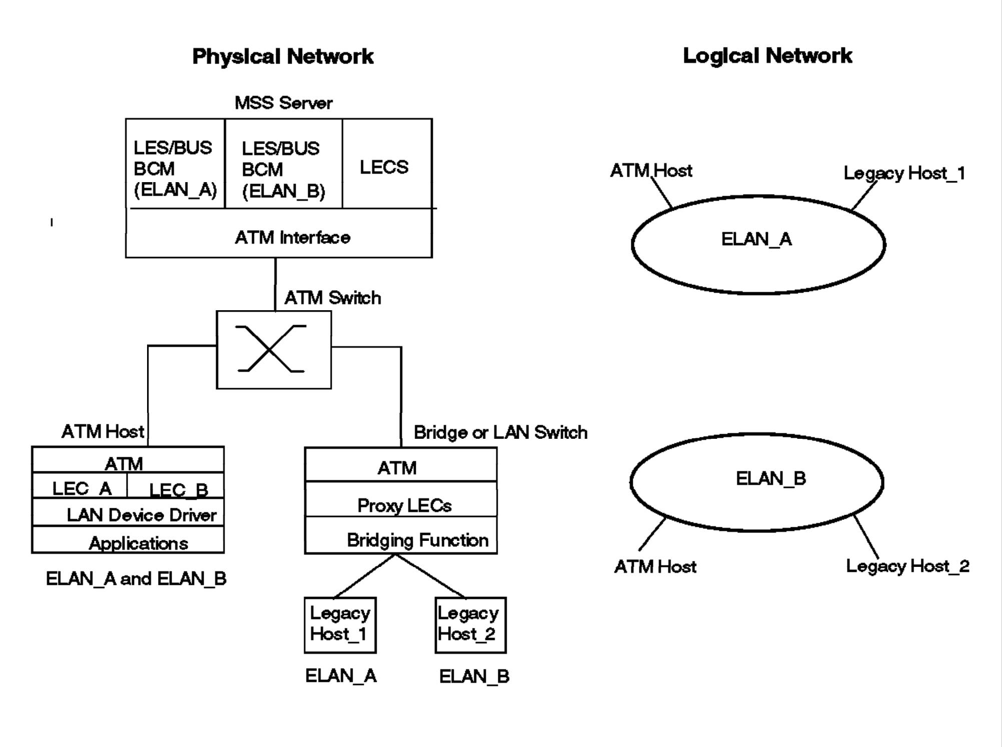


Figure 34. Physical and Logical Views of a Simple LAN Emulation Network

The LAN Emulation 1.0 specification does not specify how ELANs should be interconnected. To enable ELANs to be interconnected, the MSS Server has extensive bridging and routing functions. These bridging and routing functions are accessed by creating internal LE clients (LECs), which are assigned to the ELANs to be bridged or routed. These internal LECs convert the ATM cells to token-ring or Ethernet frames, which are then processed by the MSS′s internal bridge or router before being converted back to ATM cells to be forwarded to the destination ELAN. All members of an ELAN must be of the same type, that is, Ethernet or token-ring. ELANs of different types can, however, be bridged or routed by the MSS Server.

An LE client can only belong to a single ELAN. However, an endstation which has more than one ATM adapter (or an adapter that supports more than one LE client) can have LE clients belonging to differant ELANs.

Before going into LAN emulation in more detail, we review the basics of ATM addressing and the functions of ILMI as they relate to LAN emulation.

4.3 ATM Addresses

ATM uses 20-byte hierarchical addressing. The first 13 bytes of an ATM address are called the network prefix, and end systems obtain the network prefix component of their addresses from their adjacent switch. The next six bytes of the address are called the end system identifier (ESI), and the final byte is called the selector. End systems form their addresses by appending an ESI and selector to the network prefix provided by the switch. The selector is only significent within the end system; it is not used to route calls within the ATM network, but is used within end systems to uniquely identify called/calling parties.

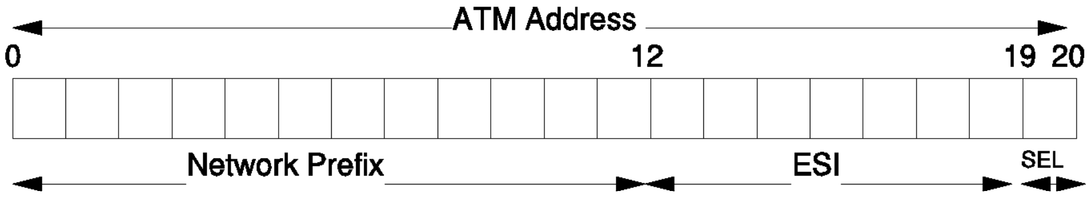


Figure 35. ATM Addresses

The network prefix and ESI components of ATM addresses must be registered with ATM switches before calls can be placed or received. If the address is not unique (that is, if it duplicates an address already registered with the switch), the switch will reject the registration. One way to guarantee a unique ATM address is to use the burned-in (universally administered) IEEE MAC address as the ESI. Each ATM interface on the IBM 8210 Nways MSS Server contains a burned-in MAC address that may be used in this manner. The IBM 8210 Nways MSS Server also allows users to configure locally-administered ESIs on each ATM interface.

4.3.1 ATM Addresses of LAN Emulation Components

In general, ATM addresses must be unique among LAN emulation components. The only exception is that a LES and BUS serving the same ELAN may share an ATM address (this is the case on the MSS Server).

LANE 1.0 Compliance

IBM has found that some vendors are violating LANE standards in their adapter implementations by requiring that the LES and BUS functions be at different ATM addresses. IBM fully complies with the standard in all of its products and has no such restriction.

LAN emulation components are configured for a particular ATM interface, and the user may decide to use the burned-in MAC address as the ESI portion of the component′s ATM address or select one of the locally-administered ESIs defined for the ATM interface. Multiple LE components may share the same ESI if they have unique selectors. By default, the configuration interface assigns each LE component a unique selector value for the configured ESI; however, the user may override this assignment and explicitly configure a particular selector value.

An ATM interface parameter determines the number of selectors per ESI reserved for explicit assignment (the remainder are avaiable for dynamic assignment by the ATM interface at run-time). LE components only use the selectors reserved for explicit assignment; by default, 200 of the 256 possible selectors per ESI are reserved for explicit assignment. Run-time selector assignment is beneficial when the user does not need to control the assigned selector (Classical IP clients are an example).

While ATM addresses must be unique among LE components, LE components may use the same ATM addresses as non-LE components such as Classical IP clients/servers.

4.4 Overview of ILMI Functions

The Interim Local Management Interface (ILMI) defines a set of SNMP-based procedures used to manage the User-to-Network Interface (UNI) between an ATM end system and an ATM switch. The following three ILMI functions are particularly relevant to LAN emulation:

1. ATM address registration
2. Dynamic determination of UNI version being run on the switch
3. Acquisition of the LECS ATM address(es)

By default, the ATM interfaces of an MSS Server use ILMI procedures to query the switch MIB in an attempt to determine the signalling version (UNI 3.0 or 3.1) being run at the switch. If the query succeeds and both ends of the ATM link have defined the same UNI version, the MSS Server configures accordingly. If both ends have defined different UNI versions, connection establishment fails. In addition to explicit definition, either or both ends can specify auto-detection of the other end′s UNI version as well. For a discussion of the resulting UNI, see 3.1.1, “User-to-Network Interface (UNI)” on page 54.

ILMI is also the method of choice for locating the LECS. The ILMI MIB at the ATM switch includes a list of LECS ATM addresses that may be retrieved by the

LECs. This is useful because the LECS ATM address(es) has to be configured at the ATM switches only, not at LECs, and there are fewer switches than LECs.

4.4.1 LAN Emulation Components

We now take a closer look at the individual components that make up an ELAN. An emulated LAN comprises the following components:

- One LAN emulation server (LES)
- One LAN emulation configuration server (LECS)
- One broadcast and unknown server (BUS)
- LAN emulation clients (LECs), such as user workstations, bridges, routers, etc.

Users connect to the ELAN via LE clients, which request services through the LAN emulation User-to-Network Interface (LUNI). The three components (LE server, LECS, and BUS) may be distributed over different physical systems or may be grouped together in one system, but logically they are distinct functions. The LAN emulation services may be implemented in ATM intermediate systems (for example, switches such as the 8260 and 8285) as part of the ATM network, or in one or several ATM end systems, such as the IBM 8210 Nways MSS Server.

As illustrated in Figure 36 on page 72, each LEC has to support a variety of VCCs across the LUNI for transport of control and data traffic.

4.4.1.1 LAN Emulation Server (LES)

The basic function of the LE server is to provide directory and address resolution services to the LECs of the emulated LAN. Each emulated LAN must have an LE server. An LE client registers the LAN address(es) it represents with the LE server. When an LE client wants to establish a direct connection with a destination LEC, it gets the destination′s MAC address from the higher-layer protocols and has to ask the LE server for the destination′s ATM address. The LE server will either respond directly (if the destination client has registered that address) or forward the request to other clients to find the destination.

An emulated token-ring LAN cannot have members that are emulating an Ethernet LAN (and vice versa). Thus, an instance of an LE server is dedicated to a single type of LAN emulation. The problems of translational bridging between different LAN types is not addressed in the ATM Forum′s LAN emulation. The IBM 8210 Nways MSS Server will, however, do translational bridging between Ethernet and token-ring for NetBIOS and SNA protocols.

The LE server may be physically internal to the ATM network or provided in an external device, but logically it is always an external function that simply uses the services provided by ATM to do its job.

Note: For a discussion of the value-adds that the MSS Server introduces in addition to the basic LES functions, see Chapter 5, “MSS Server and LAN Emulation” on page 87.

4.4.1.2 LE Configuration Server (LECS)

The LECS assigns the individual LE clients to the different emulated LANs that can exist in the ATM network. During initialization, an LE client requests the ATM address of the LE server for the ELAN to which it should be connected. An LE client is not required to request this information from the LECS; an LE server′s ATM address may be configured (system defined) in the LE client.

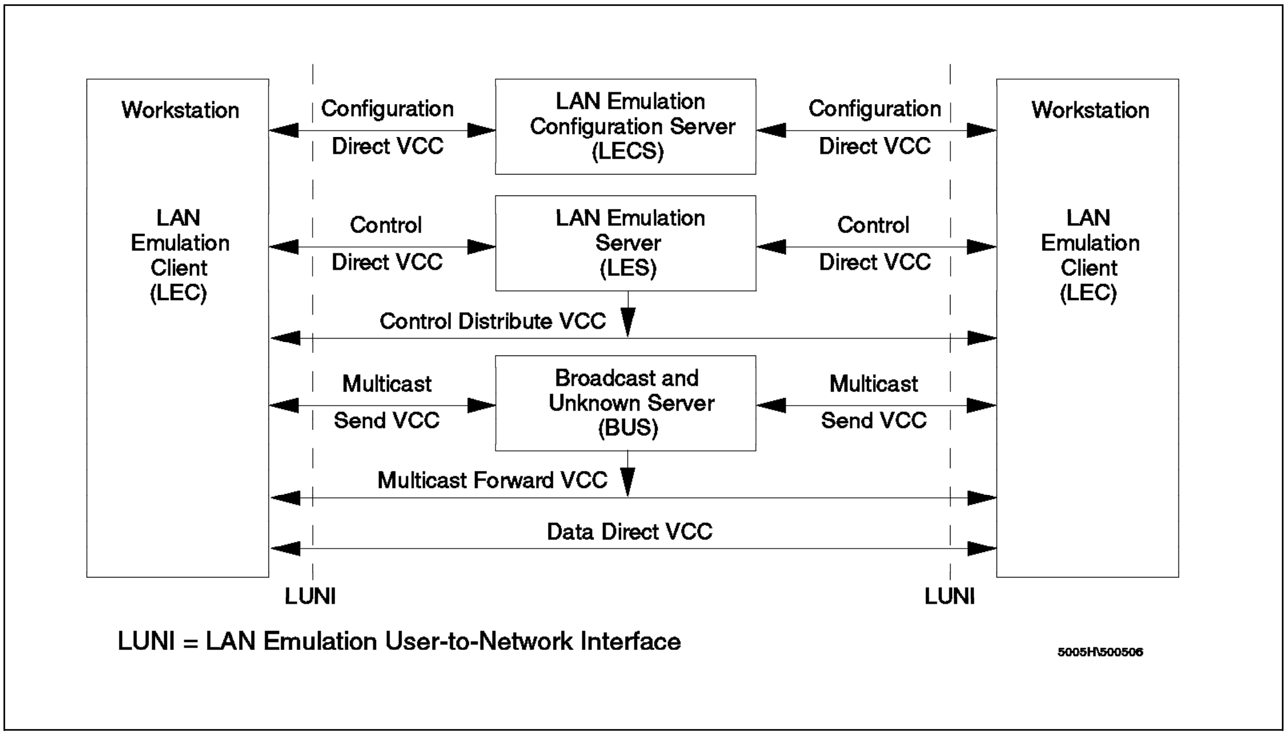


Figure 36. LAN Emulation Components

Using an LECS to assign clients to the different ELANs allows for central configuration and administration of multiple ELANs in an ATM network. The LECS could make its decision to assign an LE server, for example, based on a client′s ATM or MAC address according to a defined policy, or simply based on a system-defined database.

Note: For a discussion of the value-adds that the MSS Server introduces in addition to the basic LECS functions, see Chapter 5, “MSS Server and LAN Emulation” on page 87.

4.4.1.3 LECS Recommendations

To take advantage of MSS Server′s flexibility in creating policy-based ELANs, it is recommended that LECs be configured to use the LECS to obtain the ATM address of their LES, where possible.

4.4.1.4 Broadcast and Unknown Server (BUS)

The BUS has two main functions: (1) distribute multicast and broadcast frames to all LECs in the ELAN, and (2) forward unicast frames to the appropriate destination. A LEC sends unicast frames to the BUS if it does not have a direct connection to the LEC representing the destination. To avoid creating a bottleneck at the BUS, the rate at which a LEC can send unicast frames to the BUS is limited.

Note: For a discussion of the value-adds that the MSS Server introduces in addition to the basic BUS functions, see Chapter 5, “MSS Server and LAN Emulation” on page 87.

4.4.1.5 LAN Emulation Client (LEC)

Each workstation connecting to the ELAN has to implement the LE layer (also called LE entity), which performs data forwarding and control functions such as address resolution, establishment of the various VCCs, etc. The LE layer functions could be implemented completely in software, in hardware on a specialized LAN emulation ATM adapter, or in a combination of both. The layered structure of the LEC is shown in Figure 37.

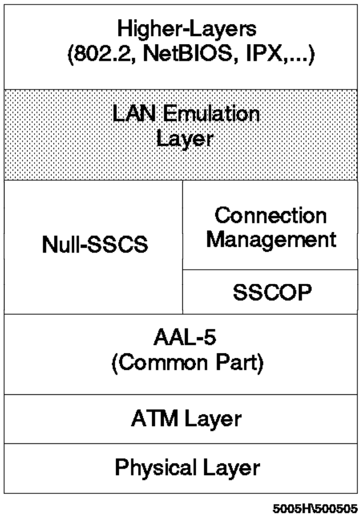


Figure 37. LAN Emulation Client Functional Layers

The LE layer provides the interface to existing higher-layer protocol support (such as IPX, IEEE 802.2 LLC, NetBIOS, etc.) and emulates the MAC-level interface of a real shared-media LAN (802.3/Ethernet or token-ring). This means that no changes are needed to existing LAN application software to use ATM services. The LE layer implements the LUNI interface when communicating with other entities in the emulated LAN.

The primary function of the LE layer is to transfer LAN frames (arriving from higher layers) to their destination either directly or through the BUS.

A separate instance of the LE layer is needed in each workstation for each different LAN or type of LAN to be supported. For example, if both token-ring and Ethernet LAN types are to be emulated within a single station, then you need two LE layers. In fact, they will probably just be different threads within the same copy of the same code, but they are logically separate LE layers. Separate LE layers would also be used if one workstation needed to be part of two different emulated LANs, both emulating the same LAN type (for example, token-ring). Each separate LE layer needs to have a different MAC address and must be attached to its own LE server, but it can share the same physical ATM connection (adapter).

4.4.2 LAN Emulation VC Connections

Data transfer in the LE system (consisting of control messages and encapsulated LAN frames) uses a number of different ATM VCCs as illustrated in Figure 36 on page 72.

4.4.2.1 Configuration and Control Connections

Control VCCs connect an LE client to the LE configuration server and the LE server, but they are never used for user data traffic. These connections may be permanent or switched and are established when an LE client connects to the ELAN.

Configuration Direct VCC

A bidirectional, point-to-point configuration direct VCC may be established between an LE client and the LECS to obtain configuration information (for example, the LE server′s ATM address).

Control Direct VCC

A bidirectional, point-to-point control direct VCC must be established (and kept active) between each LE client and the LE server. This is used for the exchange of control traffic (for example, address resolution) between the LE client and the LE server.

Control Distribute VCC

The LE server may optionally establish a unidirectional control distribute VCC to distribute control information (for example, query for an unregistered MAC address) to all LE clients connected to the ELAN. This can be a point-to-point VCC to each LE client. If the ATM supports point-to-multipoint connections, then the LE server might instead establish one point-to-multipoint VCC to all LECs (of course, the clients will be added or deleted as leaves on this point-to-multipoint tree as they enter or leave the ELAN).

Note: When using the MSS Server′s intelligent LES function (see 5.1.1, “Intelligent LES (ILES)” on page 89), two separate control distribute VCCs will be established: one to the proxy LE clients and one to the non-proxy LE clients.

4.4.2.2 Data Connections

Data connections are direct VCCs from an LE client to other LE clients and to the BUS. They are used to carry user data traffic and never carry control traffic (except for a flush message for cleanup).

Data Direct VCC

For unicast data transfer between end systems, data direct VCCs are set up through ATM signaling as bidirectional, point-to-point connections once the LE client has received the destination′s ATM address from the LE server.

As long as a data direct VCC has not been established (the protocol flows with the LE server may take some time), an LE client may send initial data frames through the BUS, but as soon as a data direct VCC is established, it has to be used and no data must be sent through the BUS. Since the LAN frames can be exchanged between two LE clients through either the BUS or using the direct VCC (that is, there are two possible paths between the clients to exchange LAN frames), careful control is needed to ensure that when the direct VCC becomes available, frames are not delivered out of sequence to the destination.

Data direct VCCs stay in place until one of the partner LECs decides to end the connection based on installation options defining relevant timeouts, etc.

Multicast Send VCC

During initialization, a LEC has to establish a bidirectional, point-to-point multicast send VCC to the broadcast and unknown server (the BUS′s ATM address is provided to the LEC by the LE server) and must keep this VCC established while being connected to the ELAN. This VCC is used by the LEC to send broadcast and multicast data frames. It is also used by the LE clients for sending unicast frames until a data direct VCC is established between the LE client and its partner. The BUS may use this VCC to send data (including multicast) to the LEC.

Multicast Forward VCC

When an LE client establishes its multicast send VCC to the BUS, the BUS learns about the new member of the ELAN. The BUS then will initiate signaling for the unidirectional multicast forward VCC to the LEC, which is used to forward data frames from the BUS to the LECs. This VCC can be either point-to-point or point-to-multipoint (of course, a point-to-multipoint VCC is more effective for multicast operations).

Every LEC must be able to receive data frames from the BUS (both over the multicast send VCC or the multicast forward VCC) but will not receive duplicates as the ATM Forum LAN emulation specification prevents the BUS from sending duplicates frames on these VCCs.

Note: When using the MSS Server′s intelligent BUS function (see 5.1.2, “Intelligent BUS (IBUS)” on page 89), two separate multicast forward VCCs will be established: one to the proxy LE clients and one to the non-proxy LE clients.

4.4.3 LE Service Operation

In operation, the LAN emulation service performs the following functions:

Initialization

During initialization, the LE client discovers its own ATM address from the ATM switch, which is needed if the client is to later set up direct VCCs. It obtains the LE server′s ATM address from the LECS and establishes the Control VCCs with the LE server and the BUS. The BUS address is provided to the LE client by the LE server.

For more details of this function, refer to 4.4.3.1, “Initialization” on page 77.

Address Registration

Clients use this function to provide address information to the LE server. A client must either register all LAN destinations for which it is responsible or join as a proxy. The LAN destinations may also be unregistered as the state of the client changes. An LE server may respond to address resolution requests if LE clients register their LAN destinations (MAC addresses, or for source routing IEEE 802.5 LANs only, route descriptors) with the LE server.

For more details on address registration, see to 4.4.3.2, “Address Registration” on page 82.

Address Resolution

This is the method used by an ATM client to associate a LAN destination with the ATM address of another client or the BUS. Address resolution allows clients to set up data direct VCCs to carry frames. This function includes mechanisms for learning the ATM address of a target station, mapping the MAC address to an ATM address, storing the mapping in a table, and managing the table.

For the server, this function provides the means for supporting the use of direct VCCs by endstations. This includes a mechanism for mapping the MAC address of an end system to its ATM address, storing the information, and providing it to a requesting endstation.

For more details on this function, refer to 4.4.3.3, “Address Resolution” on page 83.

Connection Management

In SVC environments the LAN emulation client, the LAN emulation server, and BUS set up connections between each other using UNI signaling. This function is beyond the scope of this book.

Data Transfer

To transmit a frame, the sending LE layer must do the following:

- Decide on which of its VCCs (to destination LE client or BUS) a frame is to be transmitted
- Encapsulate the frame (AAL-5 is used)

It must also decide when to establish and release data direct VCCs. To do this it may need to access the LE server for address resolution purposes.

For more details of this function, refer to 4.4.3.4, “Data Transfer” on page 84.

Frame Ordering

A sending LAN emulation client and a receiving client may have two paths between them for unicast frames, one via the BUS and one via a data direct VCC between them. A client is expected to use only one path at a time for a specific LAN destination, but the choice of paths may change over time. Switching between those paths introduces the potential for frames to be delivered out of order to the receiving client. The out-of-order delivery of frames between two LAN end systems is uncharacteristic of LANs and undesirable in an ATM emulated LAN. The Flush protocol is therefore provided to ensure the correct delivery of unicast data frames.

For more details of this function, see 4.4.3.5, “Frame Ordering” on page 84.

4.4.3.1 Initialization

The initialization of a LAN emulation client comprises the definition of the initial state and then the following five operational phases:

1. LAN emulation configuration server connect phase
2. Configuration Phase
3. Join Phase
4. Initial Registration Phase
5. Broadcast and unknown server connect phase

These phases must be completed in the specified order and if a LAN emulation client (LE client) is to achieve full interoperability, all of these phases must be completed successfully.

Initial State: The initial state of a client is an implementation issue, but certain parameters are the subject of range constraints. Parameters have a minimum and maximum value and also a default value. If any parameter falls outside of its range, the result may be a poorly functioning or possibly a non-functioning emulated LAN.

If the initialization phase terminates abnormally, the LAN emulation client must return to the initial state and inform layer management.

A full description of the LE client parameters is available in A.2, “ATM Forum LAN Emulation Client Parameters” on page 465, with the LE server parameters being detailed in A.1, “ATM Forum LAN Emulation Server Parameters” on page 465.

LAN Emulation Configuration Server (LECS) Connect Phase: In the LECS connect phase, the LE client establishes its session with the LAN emulation configuration server. Although connecting to the LECS is optional, its use is recommended. Use of the LECS enables you to centrally control which LE clients connect to which ELANs and to set LE configuration parameters. Another reason to use LECS functions to learn the LES ATM address is because the MSS Server′s LES/BUS redundancy relies on the use of LE clients connecting to an LECS first (see 5.4, “Redundant LES/BUS” on page 106).

Some LE Client drivers, however, do not support use of the LECS, in which case the LES address needs to be manually configured in the LE client and LECS functions are bypassed.

Secure ELANs

See 5.7, “Secure ELANs” on page 115 for a discussion about how the MSS Server′s LES and LECS functions can be used to verify LE clients that bypass the LECS.

The LEC may obtain the ATM address of the LECS from its adjacent switch in one of several ways. The correct order to obtain the ATM address of the LAN emulation configuration server is as follows:

1. Get LECS address via ILMI

   The LE client should issue an ILMI Get or GetNext to obtain the ATM address of the LECS for that User-to-Network Interface (UNI). The LE client then attempts to establish a configuration direct VCC to the ATM address it has received from the ILMI process. If the connection establishment fails, the LE client issues an ILMI Get or GetNext request to determine if an additional LECS ATM address is available and connects to it.

Note: See “IBM 8260 and 8285 LECS Address Support” to see how to configure the 8260/8285 to enable adjacent LE clients to learn the LECS via ILMI.

2. Use the well-known LECS address (WKA)

   If the LECS address cannot be obtained from ILMI, or if the LEC is unable to connect to any of the addresses found in the ILMI table, the LEC will attempt to connect to the following well-known address (WKA):

   x′4700790000000000000000000000A03E00000100′

   Note that the adjacent switch must be configured to map the WKA to the real address of the LECS.

   Note: See “IBM 8260 and 8285 LECS Address Support” to see how to configure the 8260/8285 to enable adjacent LE clients to use the well-known LECS address.

3. Use the LECS PVC

   If the LEC cannot establish a connection to the well-known address, then the well-known PVC of VPI=0, VCI=17 (decimal) must be used to connect to the LECS.

It is recommended that you use the ILMI procedure, unless an ATM switch or the LAN emulation device driver of an ATM adapter does not support this.

After the LEC has obtained the ATM address of the LECS, it will set up a configuration direct VCC to the LECS. The LEC then enters the configuration phase. Note that the LEC can drop this VCC whenever it wants to.

IBM 8260 and 8285 LECS Address Support: Both IBM 8260 and 8285 support the previously outlined methods of obtaining the LECS address. To put one or more addresses in the ILMI MIB of the 8260 or 8285, the following command has to be issued from the command prompt:

```
>set lan_emul configuration server inactive_wka
>Enter ATM address:
```

Multiple LECS addresses can be entered in this way. The client will attempt to connect to each address entry, in turn, until it makes a successful connection.

To address the rerouting of the well-known address to a real address, the following command must be issued from the 8285/8260 command prompt:

```
>set lan_emul configuration server active_wka
>Enter ATM address :
```

Configuration Phase: In the configuration phase, the LE client obtains the ATM address of the LE server and optionally other configuration parameters necessary to prepare the LE client to enter the join phase (see "Join Phase" on page 79).

There are the following two types of control frames used in the configuration phase:

- LE_CONFIGURE_REQUEST

  This is issued by the LE client to the LECS along the configuration direct VCC to obtain configuration information. The request contains the following information:
  - C1 The primary ATM address
  - C2 The LAN type
  - C3 The maximum data frame size
  - C4 Whether the LE client is acting as a proxy for other unicast MAC address(es)
  - C5 The name of the emulated LAN
  - C6 The local unicast MAC
  - C8 The route-descriptors (SRB bridges only)

  The local unicast MAC address is optional. ELAN name, type and maximum data frame size may be unspecified.

- LE_CONFIGURE_RESPONSE

  This is issued by the LECS in response to the LE_CONFIGURE_REQUEST. The response contains the following information:
  - C2 The LAN type
  - C3 The maximum data frame size
  - C5 The name of the emulated LAN
  - C9 The target ATM address - the address of the LE server

The configuration phase can be performed by using static parameters configured into the LE client or by using the LE configuration protocol which retrieves the parameters from the LECS. By using the LE configuration protocol, the LE client can be assigned to different emulated LANs and will learn the operating parameters of those LANs. The LECS will forward these operating parameters as well as the ATM address of the LE server to the LE client using the LE_CONFIGURE_RESPONSE.

If the LE client does not receive an LE_CONFIGURE_RESPONSE within the specified time period (configurable), it can retry until the configured number of retries is exhausted, at which time the configuration phase fails. The client will then return to the beginning of the initialization phase.

The frame formats for the configuration phase are contained within A.3, "Configuration Frame Format" on page 469.

Join Phase: In the join phase, the LE client establishes its connection with the LE server. It will determine the operating parameters of the emulated LAN and is permitted, though not required, to register one MAC address/ATM address with the LE server.

The join protocol uses the following two type of frames:

- LE_JOIN_REQUEST

This is sent by the LE client to the LE server as a request to be permitted to join an emulated LAN. It contains the following information:

- C1 The primary ATM address
- C2 The LAN type
- C3 The maximum data frame size
- C4 Whether the LE client is acting as a proxy for other unicast MAC address(es)
- C5 The name of the emulated LAN
- C6 The local unicast MAC
- C8 The route-descriptors (SRB bridges only)

The local unicast MAC address is optional. ELAN name, type and maximum data frame size may be unspecified.

- LE_JOIN_REPONSE

  This is sent by the LE server to the LE client in response to the LE_JOIN_REQUEST. It contains the following information:

  - C2 The LAN type
  - C3 The maximum data frame size
  - C5 The name of the emulated LAN
  - C14 The LE client identifier (LECID)

The LE client will set up a control direct VCC with the LE server or use a predefined control direct PVC if the LE client has been unable, or fails, to set up an SVC connection. The control direct VCC is a point-to-point, bidirectional connection. If the LE client is unable to establish either of these virtual circuits, it terminates the join phase. Once the control direct VCC is established, the LE client sends an LE_JOIN_REQUEST containing the information discussed above. Having sent the LE_JOIN_REQUEST, and if no LE_JOIN_RESPONSE has been received, the LE client will accept any request by the LE server to establish a Control Distribute VCC. However, if a response has been received, the LE client will assume no Control Distribute VCC is required and has the option to refuse an attempt by the LE server to make this connection. Further, if the LE client receives no response within the configured time allowed (the default is 120 seconds), it can retry until the configured number of retries is exhausted, at which time the join phase fails.

If the LE server does not receive the LE_JOIN_REQUEST on a new control direct VCC within a set time (the default is 120 seconds) it has the option of terminating the LE client′s membership of the emulated LAN.

When an LE_JOIN_REQUEST is received, the LE server validates the request, in which case it has the option of setting up a Control Distribute VCC to the LE client. For the request to be successful it must contain either an exact match with the LE server′s LAN type or unspecified in the LAN type (LE client variable C2). The maximum frame size (LE client variable C3) must be either unspecified or greater than or equal to the LE server′s maximum frame size.

The LE server will check for duplicate MAC addresses and/or duplicate ATM addresses already registered. If this occurs, the join request will fail and no LE client identifier will be assigned. Once the conditions are met for a successful join, the LE server will issue an LE_JOIN_RESPONSE to the LE client and the join phase of initialization is completed.

The frame formats for the join phase are contained within A.4, “Join Frame Format” on page 470.

Initial Registration Phase: When the LE client sends an LE_JOIN_REQUEST, it has the option of including one MAC address in LE client variable C6. If this occurs, the LE server will register this MAC address with the ATM address mapping as part of the join phase. The LE server will check for duplicate MAC addresses and duplicate ATM addresses already registered. If either are found, the registration request will fail.

Following a successful join phase, the LE client has the option of registering more LAN destinations using the LE registration phase and the LE client variables C6 (or route descriptor fields using variable C8 if the LE client is a source-route bridge).

Important

The registration phase is optional. However, if the registration is performed, the LE client must register all of its local unicast MAC addresses, and if the LE client is a token-ring emulated client, it must also register all its route descriptors. All registrations must be complete before the LE client reaches an operational state. Once in operational state, the LE client must not have in its variable C6 or C8 any LAN destination which has not successfully been registered with the LE server. Therefore, an LE client with only one unicast MAC address need not use the registration protocol, since it may implicitly register one MAC address during the join phase. This is because that a join with a MAC address is functionally equivalent to a join without a MAC address, followed by a register with a MAC address.

Also note that an LE client must either register all LAN destinations for which it is responsible or join as a proxy.

Broadcast and Unknown Server Connect Phase: In the broadcast and unknown server (BUS) connect phase, the LE client establishes its connection with the BUS. In order to determine the ATM address of the BUS, the LE client issues an LE_ARP_REQUEST to the LE server to resolve the all ones broadcast MAC address. The LE server will respond with the LE_ARP_RESPONSE containing the ATM address of the BUS.

The LE client uses this address to establish a bidirectional multicast send VCC. This VCC is used by the LE client to send all broadcast and multicast destination packets to the BUS. When the multicast send VCC is established, the BUS automatically establishes the multicast forward VCC. This VCC is unidirectional and can be either point-to-point or point-to-multipoint. It is used by the BUS to send multicast frames to LE clients. If the Multicast Send VCC cannot be established, the LE client may be removed from the emulated LAN.

If either party detects the release of the multicast send VCC, it will automatically release the multicast forward VCC. The LE client has the option of attempting to reestablish the connection for a configurable number of attempts. If these attempts fail, the LE client will terminate its membership of the emulated LAN.

If the LE client detects the intentional release of the multicast forward VCC, it will terminate its membership of the emulated LAN without any attempt at recovery. If it detects that this release was accidental, it may attempt to recover the connection for a configurable number of attempts after which it will terminate its membership of the emulated LAN.

In the case of the BUS detecting that the multicast forward VCC has been released, it will release that LE client′s multicast send VCC and will make no attempt to reestablish the multicast forward VCC.

4.4.3.2 Address Registration

The address registration protocol is used by an LE client wishing to register additional LAN destination and ATM address pairs not registered during the join phase.

The registration procedure can occur at any time after successfully joining an emulated LAN and is optional. The following are the four types of registration protocol frames:

- LE_REGISTER_REQUEST

  This frame is sent by the LE client to the LE server. It contains a request to register one LAN destination-ATM address pair.

- LE_REGISTER_RESPONSE

  This frame is sent by the LE server in response to the LE_REGISTER_REQUEST. It contains confirmation of a successful registration.

- LE_UNREGISTER_REQUEST

  This frame is sent by the LE client to the LE server. It contains a request to remove the registration of one LAN destination-ATM address pair.

- LE_UNREGISTER_RESPONSE

  This frame is sent by the LE server in response to the LE_UNREGISTER_REQUEST. It contains confirmation of a successful removal of the registration.

If an LE client has only one unicast MAC address, it need not use the registration protocol because every LE client can implicitly register one MAC address during the join phase. It should also be noted a join with a MAC address is functionally equivalent to a join without a MAC address followed by a register with a MAC address.

All (un)register requests are issued over the control direct VCC. The LE server will respond to (un)registration requests over either the control distribute VCC or the control direct VCC. The LE server checks for duplicate MAC addresses and duplicate ATM addresses already registered. The rules governing registrations are as follows:

- An ATM address can only be associated with one LECID.
- An ATM address/LECID mapping can have multiple MAC addresses associated with it.
- A MAC address cannot be registered by more than one ATM address.
- An ATM address/LECID mapping cannot register a MAC address already associated with another ATM address/LECID mapping.

If a registration request does not fully comply with these rules, then it will fail.

If an LE client requests the unregistration of a mapping it did not register, the LE server will send a successful response but will not actually unregister a mapping registered by another LE client.

The frame formats for address registration are held in A.5, “Registration Frame Format” on page 471.

4.4.3.3 Address Resolution

The address resolution protocol is used by the LE client to associate a MAC destination address with the ATM address of another LE client. Address resolution makes the establishment of data direct VCCs possible so that data can be transferred directly between ATM end systems.

The following are four types of address resolution frames:

- LE_ARP_REQUEST

  This frame is sent by an LE client to determine the ATM address of a given MAC address or route descriptor.

- LE_ARP_RESPONSE

  This frame is sent by the LE server in response to the LE_ARP_REQUEST to provide the information requested.

- LE_NARP_REQUEST

  This frame is sent by the LE client to advertise changes in remote address bindings.

- LE_TOPOLOGY_REQUEST

  This frame is sent by either the LE server or the LE client to indicate that network topology changes are in progress.

LE_ARP Procedure: When the LE client has a frame to transmit to an unknown MAC destination address, it issues an LE_ARP_REQUEST on its control direct VCC to the LE server.

When the LE server receives the LE_ARP_REQUEST, it can take the following actions:

1. If the MAC destination address is known to the LE server, it can issue an LE_ARP_RESPONSE. This response will contain the ATM address of the LE client responsible for the LAN destination. If the LE_ARP_REQUEST contains the broadcast MAC address, the LE server responds with the ATM address of the BUS.
2. If the LAN destination is unknown to the LE server, it will do the following:
   a. Forward the LE_ARP_REQUEST to all LE clients using either the control direct VCC or the control distribute VCC
   b. Forward the LE_ARP_REQUEST to those LE clients that registered as proxy agents using either the control direct VCC or the control distribute VCC

If the LE server has forwarded the LE_ARP_REQUEST and then receives an LE_ARP_RESPONSE from an LE client, it adds the new mapping to its LE ARP cache and forwards the response to the LE client which originated the request. The LE client adds the new mapping to its LE ARP cache.

LE_NARP Procedure: When an LE client believes that the mapping between a target LAN destination and target ATM address is no longer valid, it has the option of issuing an LE_NARP_REQUEST. This only applies to remote LAN-ATM address mappings and usually occurs because the LE client is now representing the target LAN destination at its source ATM address.

LE_TOPOLOGY Procedure: When the topology of the network changes, either the LE client or the LE server will issue the LE_TOPOLOGY_REQUEST frame to inform the other members of the emulated LAN that these changes are underway. When an LE client receives an LE_TOPOLOGY_REQUEST, it will not use any entries for non-local LAN destinations from its LE ARP cache. If the LE server receives an LE_TOPOLOGY requests it will forward this to all LE clients.

If the LE client is an IEEE 802.1D transparent bridge, it issues an LE_TOPOLOGY_REQUEST for every configuration BPDU that it issued to the BUS. These LE clients also have the option of basing the LAN emulation topology change state on the spanning tree configuration BPDU instead of those received in the LE_TOPOLOGY_REQUEST.

The frame formats for address resolution are shown in A.6, “Address Resolution Frame Format” on page 472.

4.4.3.4 Data Transfer

Once the LE client has established the ATM location of the other party, it will establish a data direct VCC with that client. The calling client may already be sending unicast data frames to the BUS to forward to the client. If this is the case, the LE client will issue an LE_FLUSH_REGISTER (see 4.4.3.5, “Frame Ordering” on page 84). The LE client will examine all frames received on any data direct VCC, and if it finds any frames with its own LECID, it will discard them. All frames will be filtered by LECID to allow only those frames required by the higher layer to be forwarded.

If the LE client receives a connection request from a client to which it already has a connection, it will accept the request, but will send frames only on the VCC that was initiated by the numerically lower ATM address. This may cause the duplicate VCC to be aged out. If the LE client detects that the control direct VCC or the control distribute VCC is released at any time other than the join phase, then the LE client must terminate its membership of the emulated LAN.

If the BUS receives a valid data frame from an LE client over a multicast send VCC, it will forward it using either the multicast forward VCC or the multicast send VCC.

Delivery of Token-Ring Frames: When the LE client has a frame to send, it examines both the frame′s destination MAC address and the routing information field to determine where to send the frame and if it is required to issue an LE_ARP_REQUEST. An LE client emulating token-ring must support address resolution of route descriptors. If the location of the target LAN destination is unknown, it must send an LE_ARP_REQUEST to the LE server. It may also send the frame to the BUS.

4.4.3.5 Frame Ordering

The LE client is able to send unicast frames to the same MAC address using the BUS and using a data direct VCC at different times. A mechanism is required to ensure that there is no possibility of delivering frames out of order by having two paths. That mechanism is known as the Flush protocol.

The flush message is a special frame identifiable as a non-data frame by having a reserved value X′FF00′ in the LAN emulation data frame header in place of the LECID of the sender.

The Flush protocol uses the following two frame types:

- LE_FLUSH_REQUEST

  This frame is sent by the LE client to either the BUS using the multicast send VCC, or another LE client using the data direct VCC. It is used to ensure that all data frames in transit on the path have reached their destination LE client.

- LE_FLUSH_RESPONSE

  This frame is sent in direct response to the LE_FLUSH_REQUEST by the called LE client. The frame is sent to the LE server using the control direct VCC or the control distribute VCC for forwarding to the LE client who initiated the flush.

Flush protocol is comprised of mandatory rules which must be applied to any component of the emulated LAN, whether they implement Flush protocol or not, and optional rules which must be applied if an LE client implements flush. The mandatory rules are discussed below.

The LE client sends LE_FLUSH_REQUEST over either the data direct VCC or the multicast send VCC. The client that receives the LE_FLUSH_REQUEST will always send the LE_FLUSH_RESPONSE to the LE server using the control direct VCC. The LE server will then forward this response to the LE client which originated the request. If the BUS receives an LE_FLUSH_REQUEST for another LE client, it will forward the request to that client using either the multicast send VCC or multicast forward VCC.

If an LE client chooses to implement flush, there are certain mandatory rules. These are discussed below.

The LE client sends an LE_FLUSH_REQUEST on either the data direct VCC or the multicast send VCC. This request must contain a transaction identifier, the source ATM address and the ATM address of the target LE client. The sending LE client cannot reuse the data direct VCC for the same LE client until it has received the LE_FLUSH_RESPONSE with matching transaction identifier. During this period, the sending LE client can either hold or discard data frames destined for a LAN destination. If the LE client does not receive a response within the required time, it will either discard any frames it is holding for that target or will send them down the old path. It can then issue another LE_FLUSH_REQUEST containing a new transaction identifier. Once the reply is received, it will send all held data frames on the new path before sending any additional frames.

The frame formats for Flush protocol are shown in A.7, “Flush Frame Format” on page 475.

4.4.3.6 Termination Phase

In the preceding section where it is indicated that the LE client will terminate the phase and all the SVCs associated with the client (including all control VCCs, data direct VCCs, and all VCCs to and from the BUS) must be released, the LE server and BUS will not attempt to reestablish the VCC for any reason.

4.4.3.7 Operation in Real Systems

In many practical LAN networks this system is going to work very well indeed. While LANs allow any-to-any data transport, typical LAN users connect to very few servers (such as communication servers, file servers, and print servers) at one time.

In this situation, the LE architecture described above can be extremely effective. After a short time, workstations will have established VCCs with all of the servers that they usually communicate with. Data transfer is then direct and very efficient. Timeouts can be set so that the switched VCCs are maintained during normal session usage.

Further details and an explanation of ATM Forum LAN emulation over ATM is available in the ATM Forum Technical Committee′s *LAN Emulation Over ATM* technical specification.

4.4.4 LAN Emulation Summary

LAN emulation provides a relatively efficient means of transporting existing LAN-based applications across an ATM network, providing them with the benefits of high-speed switched connections and scalablility. A LAN emulation system consists of the following basic components:

1. LAN emulation clients (LECs) which reside on endstations or proxy bridges/switches. These provide the interface between traditional LAN traffic based on frames and the ATM cell-based network.

2. A LAN emulation server (LES) which provides ATM to MAC address resolution.

3. A broadcast and unknown server (BUS) which forwards all broadcast traffic (and some unicast traffic) to the LE clients.

4. A LAN emulation configuration server (LECS) which assigns LES/BUS addresses to attaching LE clients. This feature is optional, but if available, provides a central administration point for the assignment of policy-based ELANs.

Chapter 5. MSS Server and LAN Emulation

The MSS Server provides support for the basic LE emulation client and server components, which are:

- LAN emulation configuration server (LECS)
- LAN emulation server (LES)
- Broadcast and unknown server (BUS)
- LAN emulation client (LEC)

All these functions are supported on the MSS Server. For an extensive description of ATM Forum-compliant LAN emulation, see Chapter 4, “ATM Forum-Compliant LAN Emulation” on page 67,

In addition, the MSS Server introduces several value-adds which enable you to increase the size of your ELANs and improve the availability and manageability. The value-adds discussed in the following sections are:

- Intelligent LES (ILES)

  The MSS Server has implemented an extension to the LES called *intelligent LES (ILES)* which may reduce the control traffic sent by the LES. For details see 5.1.1, “Intelligent LES (ILES)” on page 89.

- Intelligent BUS (IBUS)

  The MSS Server has implemented an extension to the BUS called *intelligent BUS (IBUS)* which reduces the network impact of traffic sent via the BUS. For details see 5.1.2, “Intelligent BUS (IBUS)” on page 89.

- Broadcast manager (BCM)

  The number of LAN broadcasts may limit the size of your emulated and legacy LANs. See 5.2, “Broadcast Manager (BCM)” on page 90 to see how the MSS Server has implemented an intelligent, self-learning mechanism that reduces the number of broadcasts, helping you to increase the size and efficiency of your network.

- Source route manager (SRM)

  It should be pointed out that BCM does not only reduce the number of broadcast on your ELANs, but also on the legacy LANs that are bridged to the ELANs. For token-ring networks the function of BCM can be further expanded by using the SRM function. For details see 5.3, “Source Route Manager (SRM)” on page 103.

- Redundant LES/BUS

  The LES and BUS are critical components in an ELAN as unavailability means that LE clients may be unable to communicate. 5.4, “Redundant LES/BUS” on page 106 describes the mechanism that the MSS Server has implemented that enables the use of a backup LES/BUS. Note that the redundant LES/BUS is transparent to LE clients that obtain LES addresses from the LECS.

- Redundant LECS

  When the LECS becomes unavailable, LE clients are unable to learn the ATM address of the LES and can no longer join the ELAN. 5.5, “Redundant LECS” on page 108 describes the mechanism that the MSS Server has implemented that enables the use of a backup LECS, transparent to the LE clients.

- Policy-based ELANs

  In configurations having multiple ELANs, you may want to exercise control about which LE clients join which ELANs. 5.6, “MSS Server Policies” on page 110 describes the policies implemented on the LECS to enable ELAN assignment.

- Secure ELANs

  LE clients have an option to bypass the LECS and join an ELAN by connecting to the LES directly. This may potentially violate your LECS policies. 5.7, “Secure ELANs” on page 115 discusses the mechanism that the MSS Server has implemented to enable the LES, in conjunction with the LECS, to verify the validity of join requests.

- Type/Length/Values (TLVs)

  Type/Length/Values (TLVs) enable network administrators to exercise control over certain LE client characteristics. For details, see 5.9, “Type/Length/Values (TLVs)” on page 117.

- BUS monitoring

  Specific LE clients or legacy LAN stations may be the cause of excessive BUS utilization, either because of excessive volumes on broadcast or by using the BUS for client-to-client communication. As the BUS is a shared resource for all ELAN clients, this may jeopardize the integrity of your ELAN.

  The BUS monitor functions discussed in 5.8, “BUS Monitor” on page 116, enables you to detect faulty workstations, so corrective actions can be taken.

The previous items are all enhancements to the LE services functions. They will be discussed in detail in the following section. Refer to 5.11, “Configuring LAN Emulation Services” on page 120 for configuration details.

The main reason to use the 8210 LE client functions are to enable its IP routing, IPX routing, or bridging functions. These functions are discussed in the corresponding sections within this publication, and will, therefore, not be repeated in this section. One special case of defining an LE client, however, is worth mentioning in this section as it is very useful when you use the 8210 LES/BUS to establish an ELAN to enable connectivity between ATM-attached LAN bridges, without actual use of the MSS server′s bridging function:

- Redundant transparent root bridge

  One challenge designers of large bridged (in particular, Ethernet) LANs are confronted with, is to decide which bridge is going to be the root bridge in the spanning tree and how to maximize its availability. The reason is that when a root bridge becomes unavailable, a new spanning tree has to be generated during which LAN connectivity may be interrupted for tens of seconds.

  5.10, “Redundant Spanning Tree Root Bridge” on page 118 discusses how the MSS Server can be configured to become the root bridge in your network and how to implement bridge redundancy without interrupting the spanning tree.

5.1 Intelligent LES/BUS

The MSS Server has implemented two optional functions to reduce the traffic sent by the LES and BUS. These functions are referred to as *intelligent LES (ILES)* and *intelligent BUS (IBUS)*. Each of the functions will be detailed in the following sections.

5.1.1 Intelligent LES (ILES)

LE clients learn ATM addresses associated with MAC addresses from their LAN emulation server (LES), using an LE_ARP request. The LES obtains the associated ATM address either from an ARP cache containing addresses that have been registered by ATM-attached stations or, when the MAC address is not known, by broadcasting the LE_ARP on the control distribute (PtMP) VCC.

The MSS Server has implemented an extension to the LES called *intelligent LES (ILES)*. ILES offers the option to use two separate control distribute VCCs, one to the proxy LE clients (that is, the LAN-ELAN interconnect devices) and one to the non-proxy LE clients. ILES reduces broadcast traffic from the LES, by sending the LE_ARP requests only on the proxy control distribute VCC.

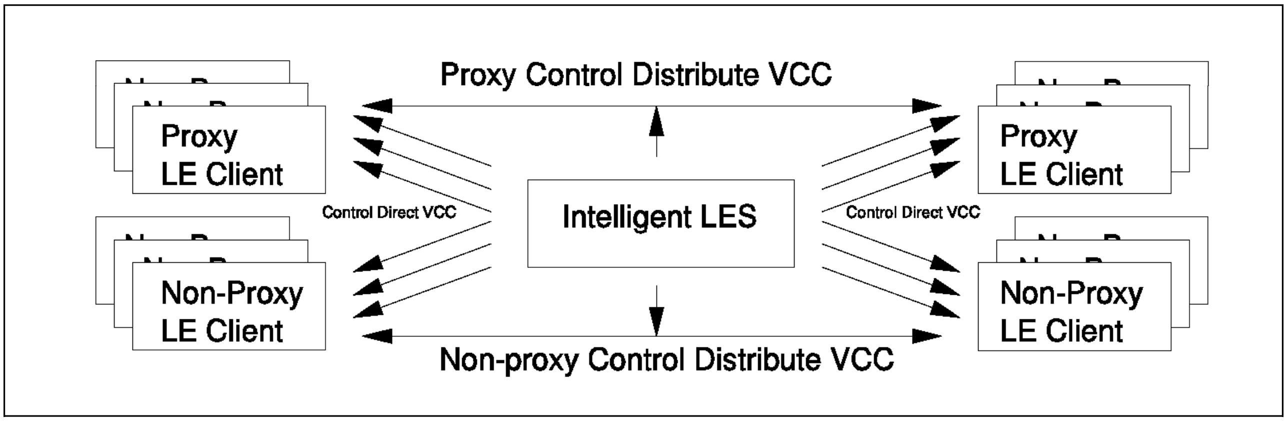


Figure 38. Intelligent LES

The ILES function is activated when selecting the *partition address resolution request forwarding domain* option in the configuration screen depicted in Figure 57 on page 127. For more details on configuring LE functions, see 5.11, “Configuring LAN Emulation Services” on page 120.

5.1.2 Intelligent BUS (IBUS)

As discussed in 4.4.1.4, “Broadcast and Unknown Server (BUS)” on page 72, the BUS has two main functions: (1) distribute multicast and broadcast frames to all LE clients in the ELAN, and (2) forward unicast frames to the appropriate destination. An LE client sends unicast frames to the BUS if it does not have yet a direct connection to the LE client representing the destination.

The MSS Server has implemented an extension to the BUS called *intelligent BUS (IBUS)*. IBUS offers the option to use two separate multicast forward VCCs, one to the proxy LE clients (that is, the LAN-ELAN interconnect devices) and one to the non-proxy LE clients. The IBUS sends multicast and broadcast frames on both multicast forward VCCs, but IBUS reduces broadcast traffic by sending unicast frames on either the proxy control distribute VCC or, when the unicast is destined for a non-proxy LE client, on the appropriate multicast send VCC.

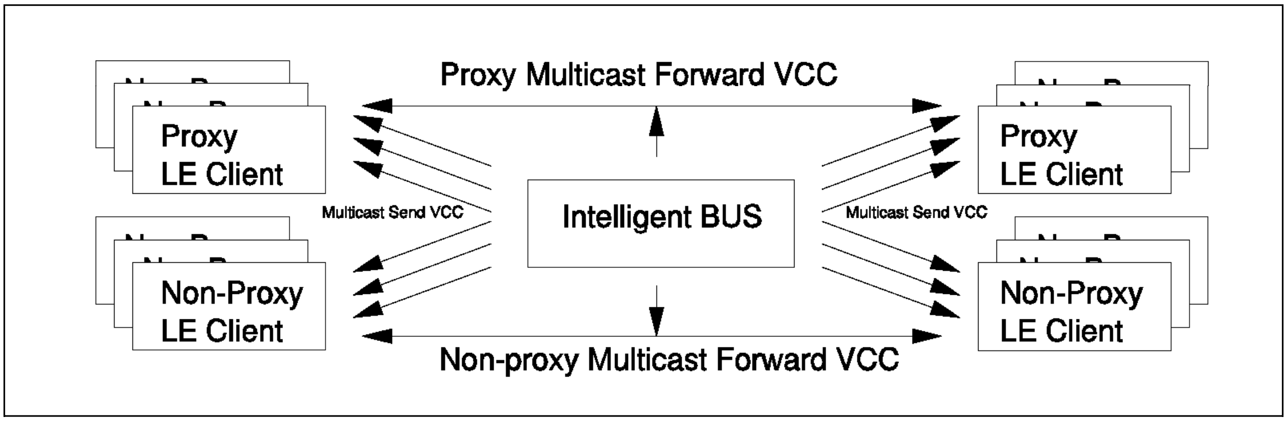


Figure 39. Intelligent BUS

One advantage of the IBUS is the reduction in client perturbation due to nuisance unicast frames (that is, unicast frames not destined for the client). Proxy clients do not receive frames destined for non-proxy clients, and non-proxy clients never receive unicast frames not destined for them. Another advantage is the reduction of network bandwidth due to nuisance frames. Disadvantages include increased BUS processing requirements and the fact that multicast/broadcast frames must be transmitted twice (once on each multicast forward VCC). In general, IBUS operation is recommended, however, this option should be disabled in configurations with source-route bridges that join the ELAN as non-proxies.

The IBUS function is activated when selecting the *partition unicast frame domain* option in the configuration screen depicted in Figure 57 on page 127. For more details on configuring LE functions, see 5.11, “Configuring LAN Emulation Services” on page 120.

Note: IBUS does not include the BCM and SRM functions.

5.2 Broadcast Manager (BCM)

Many protocols today make extensive use of broadcast frames to advertise services or to locate unknown devices. In large networks, these broadcasts can use up scarce bandwidth on LANs and, in severe cases, cause broadcast storms that can severely impact the performance of networks. In addition, these broadcasts need to be processed at the layer 3 level by each workstation adapter in order to see if they are the intended destination. This can have a negative impact on the performance of the workstations. These broadcasts are also present in ATM LAN Emulation environments as the ATM network is merely emulating a traditional token-ring or Ethernet segment. Broadcast and multicast packets are handled by the BUS as described in 4.4.1.4, “Broadcast and Unknown Server (BUS)” on page 72. Broadcasts sent to the BUS are forwarded to every LEC. Proxy LECs forward all broadcast frames onto their LAN segments.

BCM (Broadcast Control Manager) is a value-added feature from IBM which helps to control the broadcast traffic in a network. BCM′s principal goals are two-fold:

- Improve overall network performance and efficiency by reducing both network traffic and endstation processing overhead associated with filtering nuisance frames

- Enable practical deployment of larger ELANs

BCM reduces the level of broadcasts in a network in the following ways:

- Dynamically learns endstation information (for example IP addresses, IPX server and router addresses, NetBIOS names, and their associated information)
- Converts broadcast frames to unicast frames when endstations are known
- Sends unicast frames only to the interested LECs and legacy LAN endstations

BCM learns its information from its associated BUS and can be enabled and/or disabled for individual ELANs and individual protocols. Protocols supported by BCM are IP, IPX and NetBIOS. The benefits for each of these protocols is discussed in the following sections.

5.2.1 BCM and IP

When an IP station wishes to communicate with another IP device on a LAN, it needs to learn the destination MAC address first. To resolve this MAC address, it uses the Address Resolution Protocol (ARP).

To learn the destination MAC address associated with a particular destination IP address, an IP_ARP request will be broadcast to MAC address X′FFFFFFFFFFFF′. Included in the IP_ARP request are the source IP and MAC address, and the destination IP address.

Because the IP_ARP request is sent as a LAN broadcast every LAN station receives the request. However, only the endstation will recognize its IP address and return an ARP response with its required MAC address field filled in. The originating station caches this MAC address in its ARP cache so that subsequent frames can be sent without having to learn the MAC address first.

BCM for IP is activated when enabling the *“IP Broadcast Management”* option in Figure 59 on page 129. For more configuration details see 5.11, “Configuring LAN Emulation Services” on page 120.

5.2.1.1 IP without BCM

Figure 40 on page 92 illustrates the data flows when an IP datagram is sent across an ELAN without BCM enabled. It is assumed that both stations are ATM-attached and have no prior knowledge of the other station.

Note: The flows are slightly different when one or both stations are connected via a proxy.

When station A wants to send an IP datagram to station B, the IP layer in station A broadcasts an IP_ARP request to learn the MAC address of station B. The IP_ARP request contains:

- Source MAC address = MAC address station A
- Destination MAC address = X′FFFFFFFFFFFF′
- Source IP address = 9.1.100.1
- Destination IP address = 9.1.100.2

The LE layer in station A sends this frame to the BUS over the multicast send VCC.

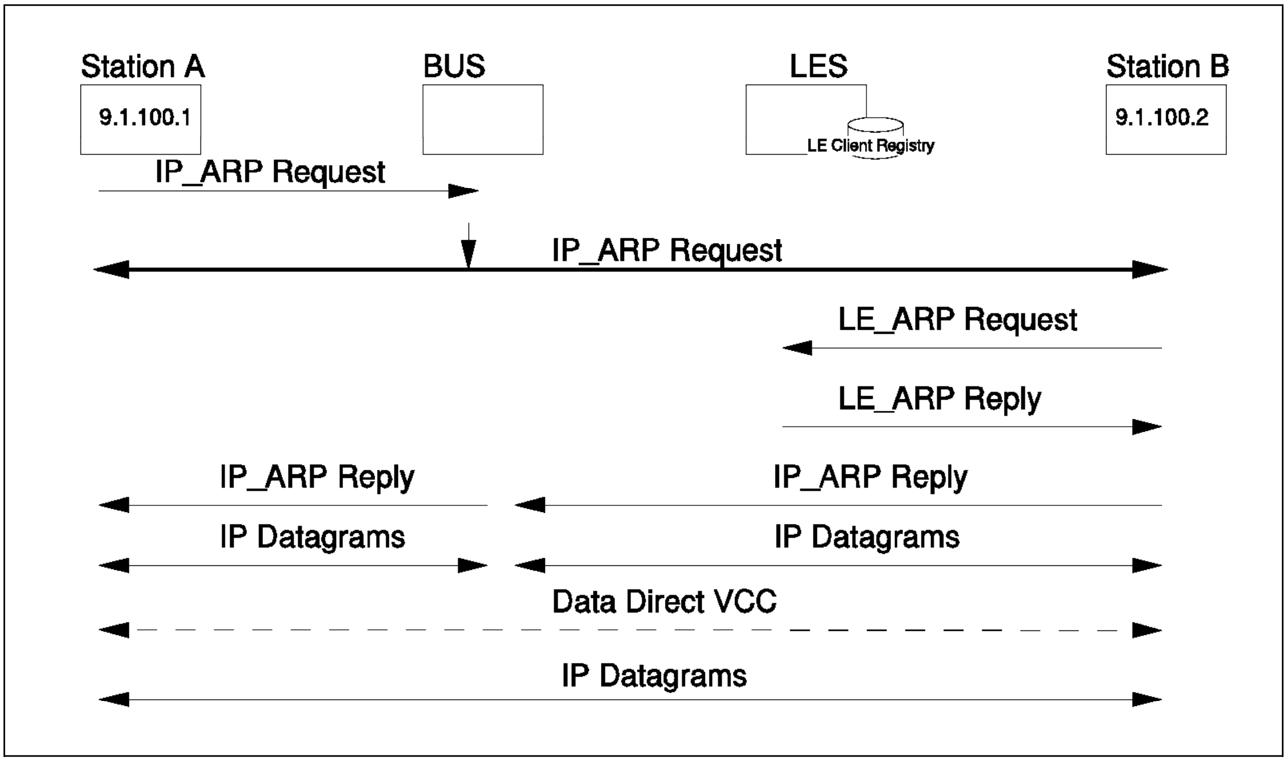


Figure 40. IP_ARP without BCM

The BUS will forward this frame to all stations on the ELAN using the multicast forward VCC. All ATM-attached stations will receive this frame and pass it to their IP layer.

Note: Proxy LECs will broadcast the IP_ARP request to their attached LAN segments.

All ATM-attached IP stations will cache the MAC address associated with station A, after which all stations, except station B which recognizes it IP address, discard the frame. Station B generates an IP_ARP reply which is passed with the following information to the LE layer:

- Source MAC address = MAC address station B
- Destination MAC address = MAC address station A
- Source IP address = 9.1.100.2
- Destination MAC address = 9.1.100.1

As the LE layer in station B has no prior knowledge of the LE client′s ATM address associated with MAC address A, station B needs to learn this ATM address first before a direct connection can be established. Until that, all data between station A and B, including the IP_ARP response, will be sent via their multicast send (PtP) VCCs to/from the BUS.

Note: The BUS transmits data destined for a legacy LAN station on the multicast forward (PtMP) VCC.

Station B learns the ATM address by sending an LE_ARP request to the LES. Because station A is ATM-attached and its MAC address has been registered, the LES returns the LE_ARP reply on behalf of station A.

Note: LE_ARP requests for stations connected via proxies are not responded, but forwarded by the LES on its control distribute VCC to all LE clients (with ILES to proxies only).

Once station B knows the ATM address of station A, it will establish a data direct VCC with station A. From here, the two stations will communicate directly with each other.

5.2.1.2 IP with BCM

In an ELAN where BCM is enabled for IP, the BUS (and therefore BCM) sees all the IP_ARP broadcast frames. IP_ARP requests and responses are scanned by BCM for the purpose of learning IP addresses, MAC addresses and LE clients of devices within this ELAN′s subnet.

When BCM receives an IP_ARP request, it caches the source IP and MAC addresses and the associated LE client. Thereafter, it looks in its table for a record of the destination IP address. If it has already cached this IP address (and the corresponding MAC address and LE client), it converts the broadcast MAC address to a unicast MAC address and sends the IP_ARP to the appropriate client on the multicast send VCC.

Figure 41 illustrates the data flows when an IP datagram is sent across an ELAN with BCM enabled. It is assumed that both stations are ATM-attached and have no prior knowledge of the other station.

Note: The flows are slightly different when one or both stations are connected via a proxy. The BCM functions, however, are identical.

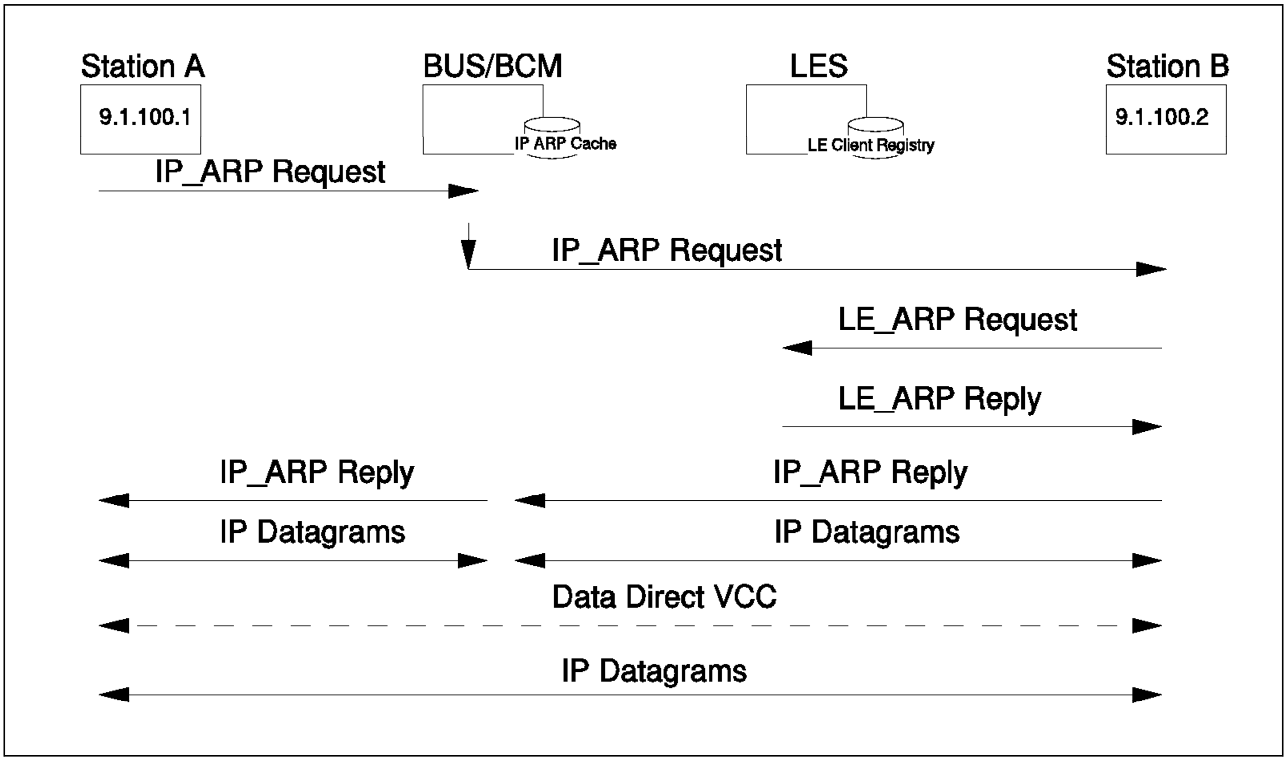


Figure 41. IP_ARP with BCM

When station A wants to send an IP datagram to station B, the IP layer in station A broadcasts an IP_ARP request to learn the MAC address of station B. The IP_ARP request contains:

- Source MAC address = MAC address station A
- Destination MAC address = X′FFFFFFFFFFFF′
- Source IP address = 9.1.100.1

- Destination IP address = 9.1.100.2

The LE layer in station A sends this frame to the BUS over the multicast send VCC. The BUS uses the information in the IP_ARP request to update its IP ARP cache for station A.

BCM and IP

When BCM receives the IP_ARP request from the BUS, the frame will be converted into a unicast frame using station B′s MAC address as the destination address. The BUS sends the IP_ARP request on its multicast send VCC.

Note: This assumes that station B has previously sent an ARP request or response and its IP address, MAC address and LE client have already been cached by BCM.

Because the IP layer in B recognizes its IP address, it returns an IP_ARP reply with the following information to the LE layer:

- Source MAC address = MAC address station B
- Destination MAC address = MAC address station A
- Source IP address = 9.1.100.2
- Destination MAC address = 9.1.100.1

As the LE layer in station B has no prior knowledge of the LE client′s ATM address associated with MAC address A, it must learn this ATM address first before a direct connection can be established. Until that, all data between station A and B, including the IP_ARP response, will be sent via their multicast send (PtP) VCCs to/from the BUS.

Note: The BUS uses the information in the IP_ARP response to update its IP ARP cache for station B.

Station B learns the ATM address by sending an LE_ARP request to the LES. Because station A is ATM-attached and its MAC address has been registered, the LES returns the LE_ARP reply on behalf of station A.

Note: LE_ARP requests for stations connected via proxies are not responded, but forwarded by the LES on its control distribute VCC to all LE clients (with ILES to proxies only).

Once station B knows the ATM address of station A, it will establish a data direct VCC with station A. From here, the two stations will communicate directly with each other.

5.2.2 BCM and IPX

Novell′s NetWare uses two protocols to advertise services and locate network resources. These are:

- IPX-RIP (Routing Information Protocol)

  IPX-RIP is used by NetWare routers to exchange routing information. In addition its enables IPX workstations to learn the optimum route before connecting to a server.

  RIP messages are broadcast by:

  - IPX routers

        - Periodically (every 60 seconds)
        - As a result of network or router state (up/down) changes

    - IPX workstations

        - To learn a route to a particular server

          Note: The reply is returned as a unicast message

- SAP (Service Advertisement Protocol)

  The service advertisement protocol enables IPX servers to broadcast their services throughout the network. In addition its enables IPX workstations to learn their nearest server.

  Note: The term *server* is used to denote any service-oriented process that may be running on a server, router or workstation.

  SAP messages are broadcast by:

    - IPX servers

        - Periodically (every 60 seconds) using a SAP Information_Broadcast

    - IPX workstations

        - To find the nearest server using a SAP NEAREST_SERVICE request

          Note: The SERVICE_RESPONSE is returned as a unicast message.

In large IPX networks, these protocols can generate large amounts of overhead and severely impact network performance. In general, RIP and SAP information only needs to be shared between network routers and servers. Client workstations do not need to see these broadcasts. BCM limits RIP/SAP broadcasts in the following ways:

- By learning the location of IPX routers and servers
- By converting broadcast frames to unicast frames and forwarding only to these learned entities

Note: Quiet devices (ones which do not advertise their presence) but which still need to receive RIP/SAP updates, will need to be configured as static entries in BCM.

BCM for IPX is activated when enabling the IPX Broadcast Management option in Figure 59 on page 129. For more configuration details see 5.11, "Configuring LAN Emulation Services" on page 120.

5.2.3 IPX without BCM

All broadcast RIPs and SAPs in an ELAN are sent via the BUS. When BCM is disabled for IPX, these packets are forwarded on the multicast forward VCC in the normal way. Proxy LECs (such as bridges and switches) will broadcast these packets on all of their local segments. These broadcasts limit the size of the ELANs which can be built in this environment.

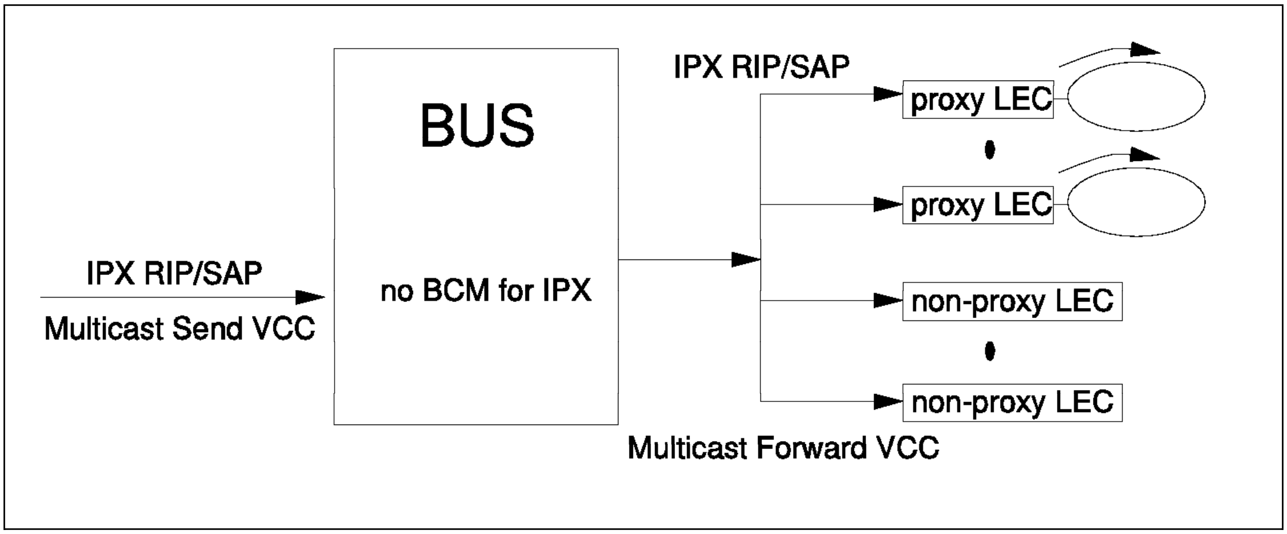


Figure 42. IPX without BCM

5.2.4 IPX with BCM

When BCM is enabled for IPX in a particular ELAN, BCM monitors the broadcast RIP and SAP traffic on the BUS and learns the location of IPX servers and routers in that ELAN. Since only servers and routers need to see the RIP/SAP broadcast traffic, BCM converts these broadcast packets to unicast packets (using the LEC ATM address, MAC address and IPX network address in its cache).

These unicast packets are then sent to each of the servers/routers in turn on the BUS′s multicast send VCCs. In the case of workstations on traditional LANs, these packets will be forwarded only to the servers and routers on these LANs.

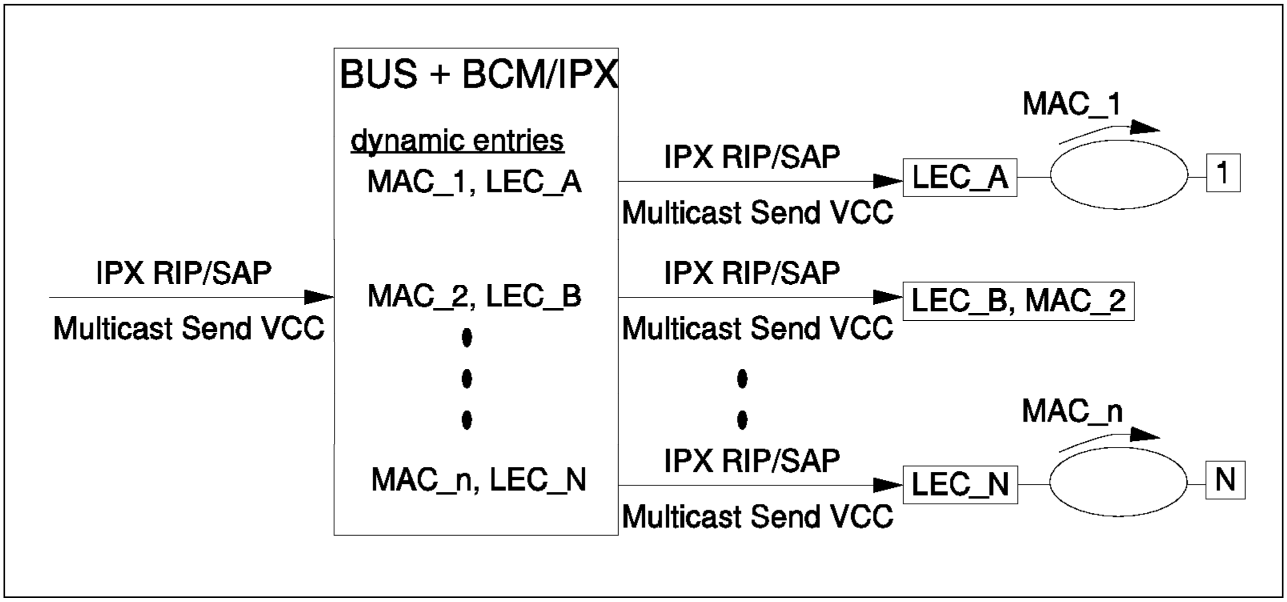


Figure 43. IPX with BCM

Care must be taken when deciding whether or not to implement BCM in an IPX environment. Assume you have a legacy LAN with 10 servers. If the LAN is ATM-attached via a proxy LEC, activating BCM for IPX results in the proxy LEC receiving 10 unicast RIP/SAP updates for each RIP/SAP update that is processed by BCM. Due to the extra processing on the 8210 and the increased network load, congestion might result. For this reason, BCM for IPX is automatically disabled when the number or learned servers/routers reaches 50 (see also the discussion in 5.2.4.2, “Static IPX/BCM Definitions - Servers Farms” on page 98).

5.2.4.1 Static IPX/BCM Definitions - Passive Device

As explained in the previous section; BCM learns the location of IPX servers by listening to RIP and SAP broadcasts. Instead of broadcasting RIP/SAP updates on the multicast forward VCC, it sends a copy of the RIP/SAP update as a unicast message on one or more multicast send VCC(s).

BCM might introduce problems when you are using a passive device that does not actively participate in the RIP/SAP traffic (one might think of a network management station or network traffic diagnosing tool that only listens to RIPs and SAPs and does not transmit any). Due to the transformation of multicast traffic in unicast messages, these stations might cease to receive the RIP/SAP traffic.

To overcome this problems up to three static BCM/IPX entries can be defined (for details on configuration see 5.11, “Configuring LAN Emulation Services” on page 120). Each entry consists of a MAC address and an LE client′s ATM address. When BCM transforms multicast IPX traffic into multiple unicast messages, copies will be sent to the passive (static) stations and the active (dynamically) learned servers.

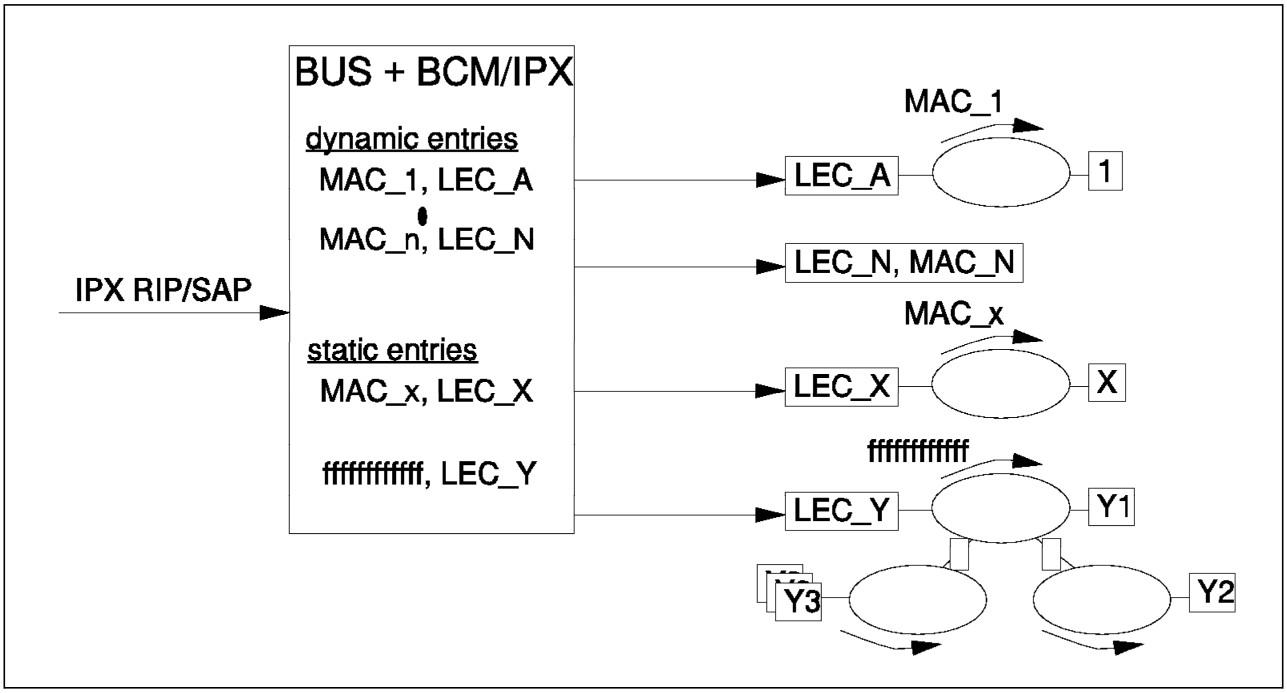


Figure 44. Active and Passive Servers

Note: The (up to) three static entries are added to the number of dynamic entries. If the total number exceeds 50, BCM/IPX is disabled.

5.2.4.2 Static IPX/BCM Definitions - Servers Farms

When defining a static entry an LE client′s ATM address, and a valid 6-byte MAC address needs to be defined. Figure 44 on page 97 depicts what will happen if you define a MAC address equal to X′FFFFFFFFFFFF′. When BCM forwards a RIP/SAP message to LE client Y (the proxy client associated with the all-ones MAC address within the static definition), LE client Y will broadcast the RIP/SAP message onto its legacy LAN. As a result all (passive and active) servers on the LAN will receive the RIP/SAP message.

This facility becomes very useful when you have a large number of servers connected via the same LEC. Without using a static BCM entry, BCM will sent a separate frame to each server, resulting in duplicate frames, and, if the number of 50 cached servers is exceeded, disabling of the BCM function for IPX. Defining a static all-ones entry results in lesser traffic. The servers that are reached via a specific all-ones static entry, is called a *servers farm*.

Note: In the case that an LE client is defined in a static entry using the LAN broadcast address (X′FFFFFFFFFFFF′), and this LE client is also connected to an active server (learned by BCM/IPX), BCM sends only the all-ones multicast RIP/SAP update.

5.2.5 BCM and NetBIOS

NetBIOS is considered to be a broadcast abusive protocol and therefore an excellent candidate for BCM.

NetBIOS uses name registration and resolution procedures that require NetBIOS applications to make heavy use of (connectionless) datagrams as well as connection-oriented LLC services, and frequently broadcast them to multiple NetBIOS endstations at the same time.

NetBIOS application resources are defined by names that are 16 bytes in length. A unique name may only exist at one endstation in a network, while a group name may be shared by multiple endstations.

An application may communicate with another application using datagrams, or, for the connection-oriented traffic, after a session has been established first. Sessions may only exist between applications using unique names, whereas group name traffic is datagram based. Multiple concurrent sessions may exist on a single LLC connection between NetBIOS endstations, and even between two applications. Only one LLC connection may carry NetBIOS traffic between any two endstations. The endstations set up the LLC connection when the first session is established and disconnect it when the last session ends.

When an application becomes active, it calls the other application to establish the session between them, sends application data, and then ends the session. The different phases in this scenario are:

1. Name Registration

   When an application becomes active, it broadcasts its name within a NetBIOS ADD_NAME_QUERY to all endstations in the network, to make sure that no other application is using the same name. This frame is addressed to the NetBIOS functional address. No reply indicates that there is no name collision. NetBIOS repeats the broadcast several times to ensure that the datagram is received by all the stations.

2. Name Search

At some time after the name-registration phase, the NetBIOS application issues a NetBIOS NAME_QUERY to detect the location of a partner application. This frame is broadcast to the NetBIOS functional address and, therefore, received by all the NetBIOS stations. Only the station with the specified name responds using the NetBIOS NAME_RECOGNIZED frame.

3. Connection Establishment

   Once the first endstation has learned the destination address on the station in which the destination NetBIOS application is located, it establishes an LLC connection.

4. Connected

   When connected, endstations exchange frames carrying application data.

5. Disconnect

   When either application decides to end the session between endstations, it sends a certain LLC frame that signals session end.

When an application using a group name starts, NetBIOS broadcasts an ADD_GROUP_NAME_QUERY datagram to see if any NetBIOS applications are using the new name as a unique name. If another application detects a name collision, it broadcasts a NAME_IN_CONFLICT datagram.

NetBIOS applications make heavy use of datagrams for sending application data. On a LAN, these datagrams are LLC frames that are normally broadcast to the NetBIOS group address. They may be sent to and from both unique and group addresses.

Note: Due to the fact that group membership is not broadcasted repeatedly, BCM will not operate on group names.

5.2.5.1 NetBIOS without BCM

Figure 45 on page 100 illustrates the data flows when NetBIOS datagrams are sent across an ELAN without BCM enabled. It is assumed that both stations are ATM-attached and have no prior knowledge of the other station.

Note: The flows are slightly different when one or both stations are connected via a proxy.

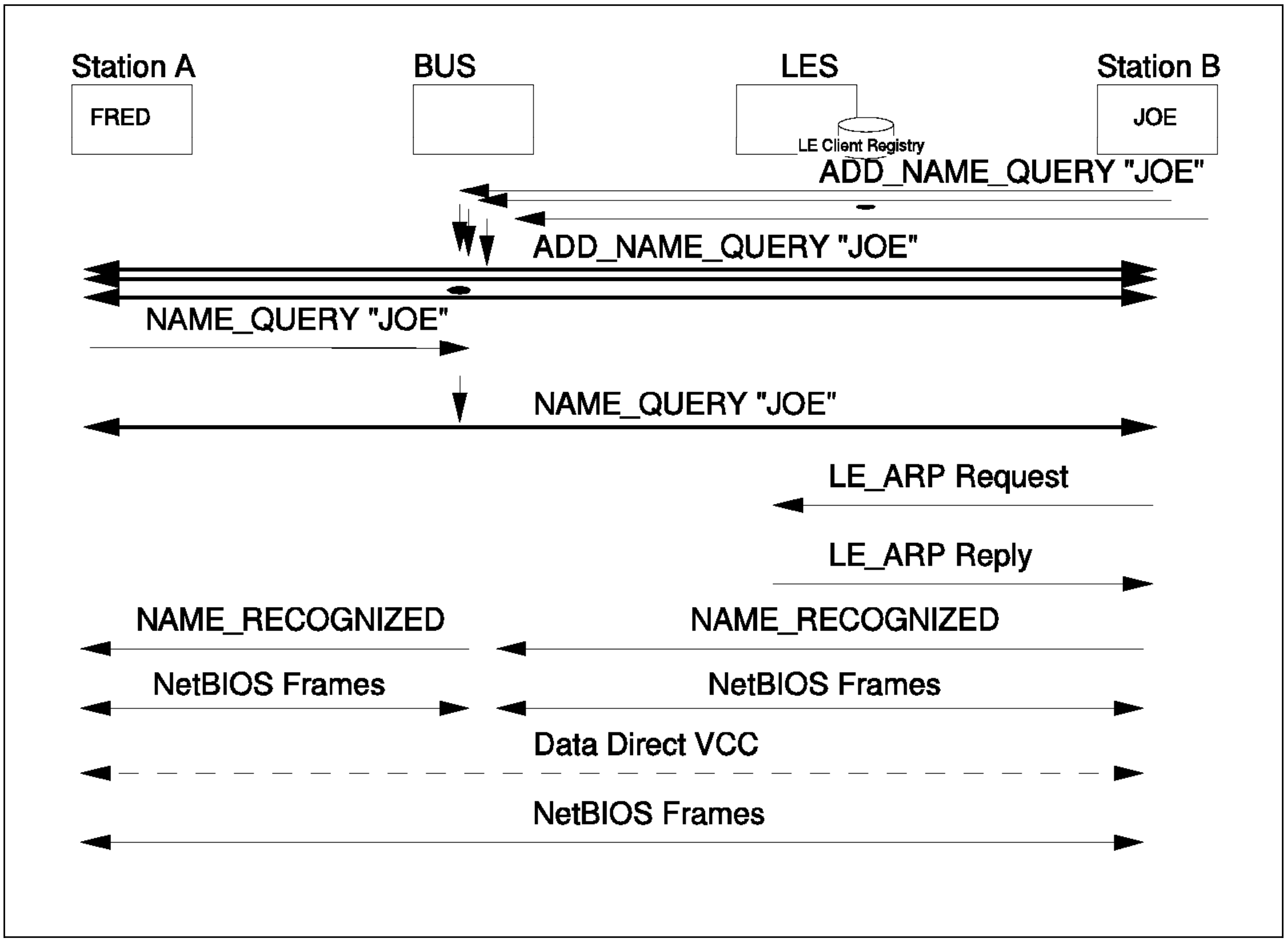


Figure 45. NetBIOS without BCM

Before stations can communicate, their names must be registered using ADD_NAME_QUERY datagrams. ADD_NAME_QUERYs datagrams sent by station B contain the following information:

- Source MAC address = MAC address station B
- Destination MAC address = X′C00000000080′
- NetBIOS name = JOE

ADD_NAME_QUERY datagrams are sent multiple times (typically six). A similar sequence is done by station A.

The LE layer sends the ADD_NAME_QUERY datagrams to the BUS over the multicast send VCC. The BUS will forward this frame to all stations on the ELAN using the multicast forward VCC. All ATM-attached stations will receive the datagram, and pass it to their NetBIOS applications to register the name.

Note: Proxy LECs will broadcast the ADD_NAME_QUERY datagram to their attached LAN segments.

When FRED wants to communicate with JOE, it performs a name search by broadcasting a NetBIOS NAME_QUERY to the NetBIOS functional address. This request will be passed to its LE layer and forwarded to the BUS over the multicast send VCC. The BUS will forward this frame to all stations on the ELAN using the multicast forward VCC.

All ATM-attached stations will receive this datagram. All except station B will discard the frame. JOE (in station B) will create a NetBIOS NAME_RECOGNIZED frame destined for the MAC address of station A, and pass it to its LE layer.

As the LE layer in station B has no prior knowledge of the LE client′s ATM address associated with MAC address A, it must learn this ATM address first before a direct connection can be established. Until that, all data between station A and B, including the NAME_RECOGNIZED response, will be sent via their multicast send (PtP) VCCs to/from the BUS.

Note: The BUS transmits data destined for a legacy LAN station on the multicast forward (PtMP) VCC.

Station B learns the ATM address by sending an LE_ARP request to the LES. Because station A is ATM-attached and its MAC address has been registered, the LES returns the LE_ARP reply on behalf of station A.

Note: LE_ARP requests for stations connected via proxies are not responded, but forwarded by the LES on its control distribute VCC to all LE clients (with ILES to proxies only).

Once station B knows the ATM address of station A, it will establish a data direct VCC with station A. From here, the two stations will communicate directly with each other.

5.2.5.2 NetBIOS with BCM

Figure 46 on page 102 illustrates the data flows when NetBIOS datagrams are sent across an ELAN with BCM enabled. It is assumed that both stations are ATM-attached and have no prior knowledge of the other station.

Note: The flows are slightly different when one or both stations are connected via a proxy. BCM functions, however, are identical.

Before stations can communicate, their names must be registered using ADD_NAME_QUERY datagrams. ADD_NAME_QUERY datagrams sent by station B contain the following information:

- Source MAC address = MAC address station B
- Destination MAC address = X′C00000000080′
- NetBIOS name = JOE

ADD_NAME_QUERY datagrams are sent multiple times. BCM uses the information in the ADD_NAME_QUERY to update its NetBIOS name cache information. BCM filters all datagrams except the first one. A similar sequence is done by station A.

The LE layer sends the ADD_NAME_QUERY to the BUS over the multicast send VCC. The BUS will forward this frame to all stations on the ELAN using the multicast forward VCC. All ATM-attached stations will receive this frame, and pass it to their NetBIOS applications to register the name.

Note: Proxy LECs will broadcast the ADD_NAME_QUERY datagram to their attached segments.

When FRED wants to communicate with JOE, it performs a name search by broadcasting a NetBIOS NAME_QUERY to the NetBIOS functional address. This request will be passed to the LE layer and forwarded to the BUS over the multicast send VCC.

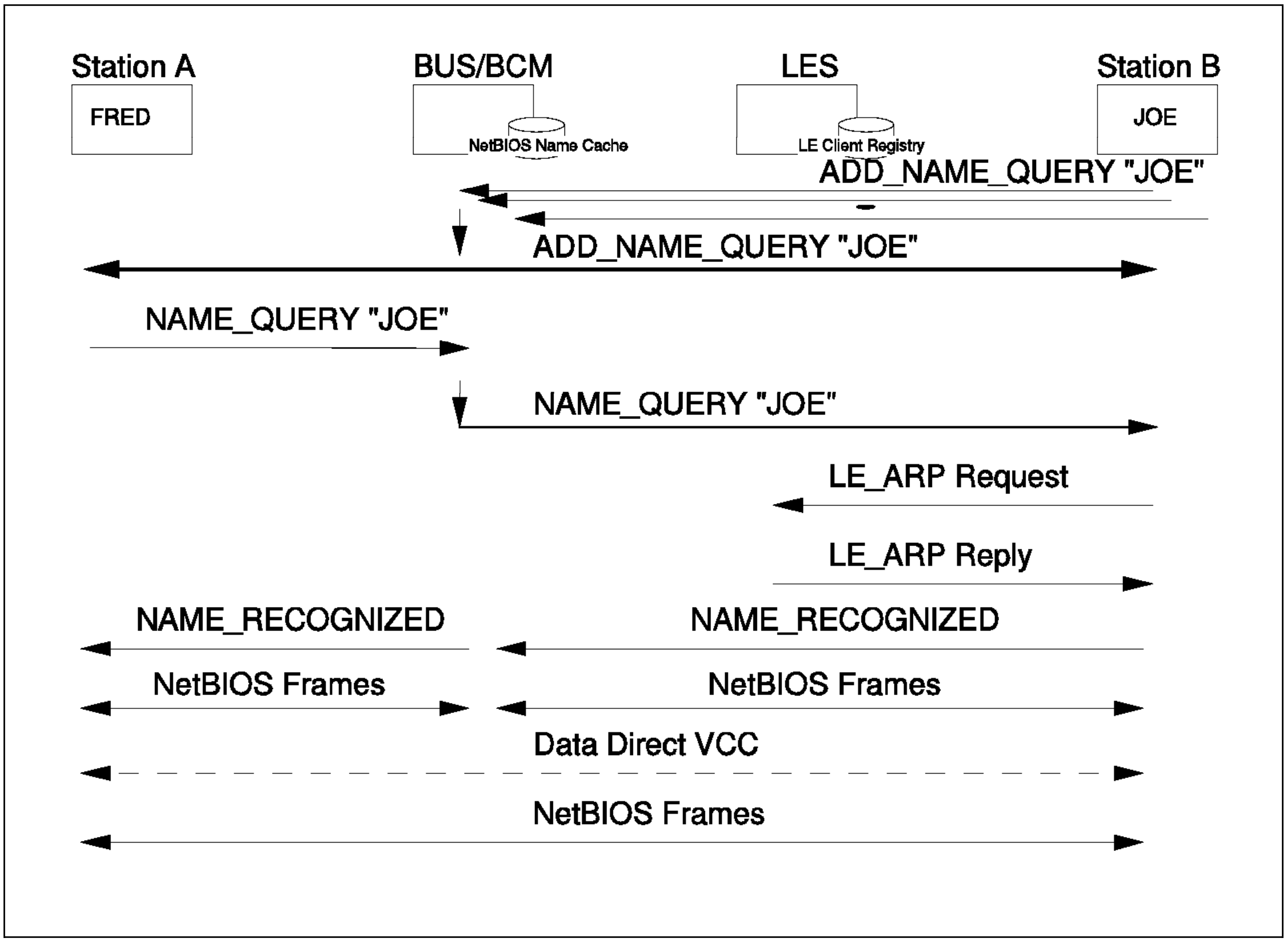


Figure 46. NetBIOS with BCM

BCM and NetBIOS

BCM will intercept the NAME_QUERY frame, and using the information learned from the ADD_NAME_QUERY frames before, convert the multicast frame into an unicast frame (using MAC address B), and forward it to station B using the multicast send VCC.

When station B receives the NAME_QUERY request application FRED will create a NetBIOS NAME_RECOGNIZED frame destined for the MAC address of station A, and pass it to its LE layer. As the LE layer in station B has no prior knowledge of the LE client′s ATM address associated with MAC address A, it must learn this ATM address first before a direct connection can be established. Until that, all data between station A and B, including the NAME_RECOGNIZED response, will be sent via their multicast send (PtP) VCCs to/from the BUS.

Station B learns the ATM address by sending an LE_ARP request to the LES. Because station A is ATM-attached and its MAC address has been registered, the LES returns the LE_ARP reply on behalf of station A.

Note: LE_ARP requests for stations connected via proxies are not responded, but forwarded by the LES on its control distribute VCC to all LE clients (with ILES to proxies only).

Once station B knows the ATM address of station A, it will establish a data direct VCC with station A. From here, the two stations will communicate directly with each other.

BCM for NetBIOS is activated when enabling the *NetBIOS Broadcast Management* option in Figure 59 on page 129. For more configuration details see 5.11, “Configuring LAN Emulation Services” on page 120.

5.3 Source Route Manager (SRM)

Token-ring endstations normally use source routing bridging (as opposed to transparent bridging) to find the most efficient route through a bridged network. Source routing makes use of two types of broadcast frames: all routes explorer (ARE), and, spanning tree explorer (STE). In large token-ring networks (or emulated LANs), these explorer frames can generate excessive traffic, leading to congested LAN segments.

Source route manager is an additional feature of BCM for token-ring (802.5) ELANs. When enabled, this feature will further process frames managed by BCM and, whenever possible, will transform ARE and STE frames to specifically routed frames (SRF). Such frames would no longer need to be transmitted onto each ring in the bridged network. The token-ring topology behind each LEC is learned by recording the routing information field (RIF) of frames received by the BUS.

Note

Source Route Manager only processes frames that have already been processed by BCM. Therefore, only IP, IPX and NetBIOS frames are supported.

The following example demonstrates how SRM learns information about endstations and uses this information to convert STE and ARE frames to specifically routed frames. An IP_ARP request was chosen in this case (see Figure 47 on page 104).

5.3.1 Details on SRM

Figure 47 on page 104 depicts the operation of source route manager. It depicts a LAN-attached station A that wants to communicate with LAN-attached station X, via an ELAN controlled by the MSS Server′s LES/BUS function. BCM and SRM are enabled.

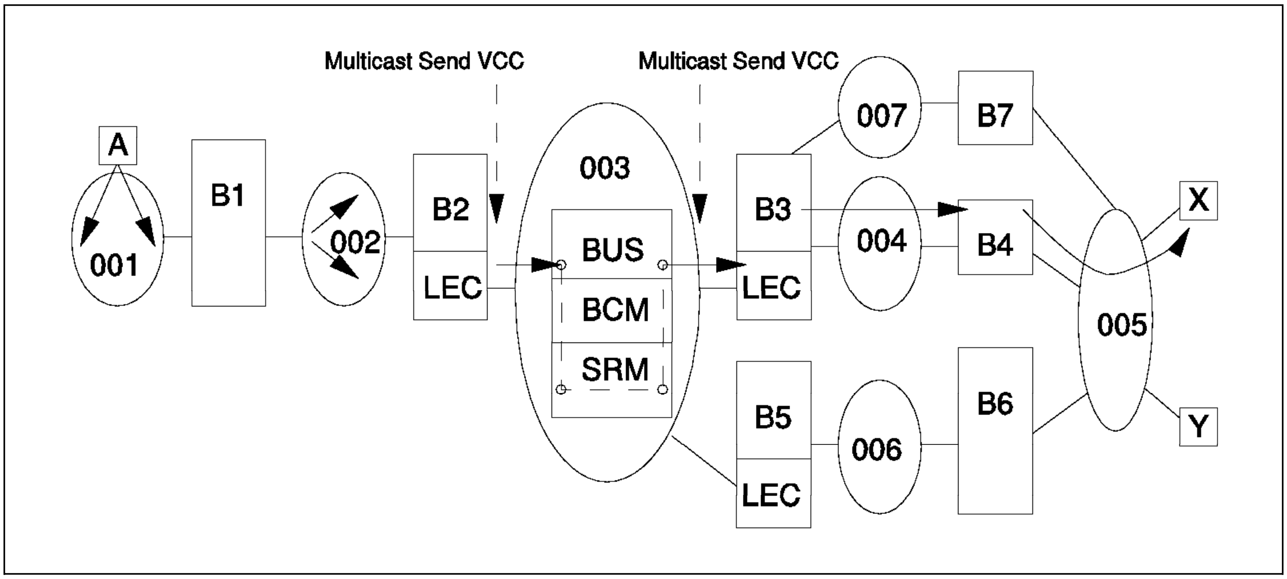


Figure 47. Source Route Management

Before station A can communicate with station X, it requires the remote MAC address. To learn this MAC address, station A broadcasts an IP_ARP request on its local LAN segment. This frame will be sent with the routing information indicator (RII) bit off to indicate that it does not have a RIF, and should not be bridged.

Since station X is on a remote LAN, the ARP request will initially time-out. Station A will generate a new ARP request, with RII on, and a RIF indicating that this is an ARE frame. Bridges B1 and B2 will forward this frame while adding routing information to the RIF in the process. The LE client function in bridge B2, seeing that the frame is an explorer frame will forward it to the BUS on its multicast send VCC.

With BCM/SRM enabled, BCM learns the following information from the explorer frame:

1. Station A layer three protocol address. Associated with this address are:
   - Station A′s MAC address
   - The ring number (001) to which station A connects
   - Proxy LE client B2 connecting ELAN and legacy LAN

   BCM will cache this information.
2. Proxy LE client B2 and ring number 001. Associated with these is:
   - The routing information field (RIF) from the ELAN to ring number 001 (003:B2,002:B1,001).

   SRM will cache this information.

Figure 48 on page 105 depicts the information cached. For a discussion how the various items relate, see the discussion in 5.3.1.1, "Best Path" on page 105.

Assuming that BCM has already learned about station X, BCM/SRM has similar information protocol and connectivity information for X:

1. Station X layer three protocol address. Associated with this address are:
   - Station X′s MAC address

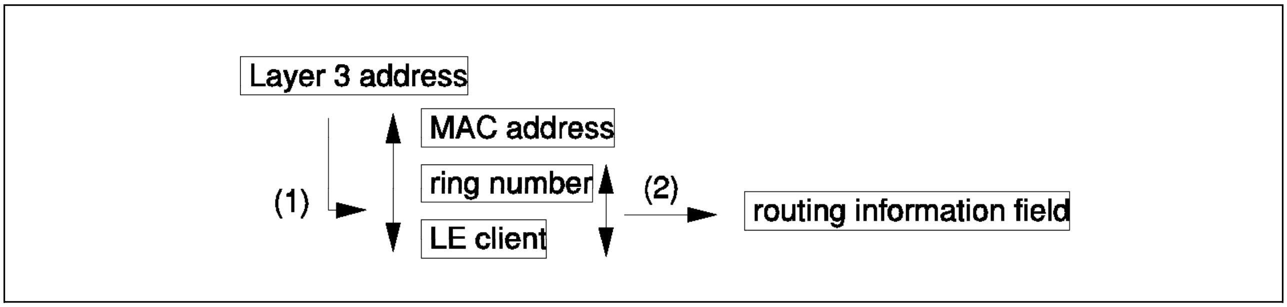


Figure 48. Entries Cached

- The ring number (005) to which station X connects
- Proxy LE client, for example, B3 connecting ELAN and legacy LAN

2. Proxy LE client B3 and ring number 005. Associated with these is:
    - The routing information field (RIF) from the ELAN to ring number 005, for example (003:B3,004:B4,005).

BCM will examine the ARP request′s destination layer three (that is, IP) address and check for a record of it in its database. As a match will be found for station X, BCM will transform the broadcast into a unicast frame. The remainder of the RIF field will be inserted to enable the frame to be forward to the LAN segment on which station X resides. Note that if SRM cannot find a record of the RIF in its database the frame will still be sent as a unicast frame, but remain an ARE frame.

The BUS will then send the (unicast) frame to proxy LE client B3 using its multicast send VCC. B3 and B4 will source route the frame to its final destination.

Note: BCM/SRM is only involved in finding the best route between two stations; once a connection between A and X has been established no BCM/SRM processing is required (all unicast data is sent specifically routed between B2 and B3, bypassing BCM/SRM).

5.3.1.1 Best Path

As can be see from Figure 47 on page 104 multiple routes exist to forward traffic from the emulated LAN (ring number 003) to station X. Therefore, BCM/SRM has multiple choices when (1) selecting the proxy LE client to connect to the legacy LAN, and (2) deciding on the RIF from the proxy client towards the LAN segment to which the destination station connects.

Layer 3 Address Caching: For every frame sent to the BUS of a protocol for which BCM is enabled, BCM will update the information {MAC address, LE client, ring number} associated with the (source) layer three address within the frame, unless the cached entry is less than 2.0 seconds (*hold-down timer*) old. The latter preserves the first protocol mapping received in a rapid series of frames from a given station.

For example, if station X in Figure 47 on page 104 sends an all route explorer frame, BCM will receive multiple copies of the frame, however, the second and following frame(s) will not result in an update of the layer three information.

It should be pointed out that each different layer three protocol address results in a separate entry. Therefore, the LE client associated with station X may be

B3, while B5 is associated with station Y. As a result, traffic from station A to X would traverse a different path than traffic from station A to Y.

Layer three caching will be done for the protocols (IP, IPX, NetBIOS) for which BCM/SRM has been enabled. Because BCM dynamically learns the location of network entities, an aging mechanism is used to remove information that has not been refreshed recently. The aging timers are configurable on a protocol basis.

RIF Caching: Every time BCM/SRM processes a frame it will update the routing information field (RIF) associated with {LE client, ring number} if the new RIF is as good or better than the RIF already cached. For this BCM/SRM uses two criteria; minimum number of hops and maximum frame size supported by the bridges along the route. A route with less hops or the same number of hops and a larger maximum frame size is always considered more optimal.

RIF caching update will be done for any protocol (IP, NetBIOS, IPX, SNA, DecNet, etc.) unicast or broadcast data processed by the BUS. The SRM aging timer for the RIF is two minutes.

SRM is activated when enabling the Source Route Management for 802.5 ELAN option in Figure 59 on page 129. For more configuration details see 5.11, “Configuring LAN Emulation Services” on page 120.

5.4 Redundant LES/BUS

A major requirement of today′s networks, carrying mission-critical data, is the ability to provide fault-tolerance for every major element in the network. A limitation of the existing ATM Forum-compliant LAN emulation standard (Version 1.0), is the lack of a redundancy mechanism for the LES/BUS function. Failure of a LES/BUS will result in all members of the ELAN being serviced by that LES/BUS to be disconnected and unable to re-establish their VCCs.

While the ATM Forum is currently working on a distributed LE service model which will address this issue in the future, IBM 8210 Nways MSS Server provides a solution to this problem today.

5.4.1 Redundant LES/BUS Operation

When a LES/BUS pair is being configured, the user has the option to enable redundant operation. If redundancy is enabled, the LES/BUS must be designated as a primary or as a backup server for that particular ELAN.

During initialization, the primary LES/BUS initiates the establishment of a redundancy VCC to the backup LES/BUS. The presence of the redundancy VCC indicates that the primary LES/BUS is operational. If an error is encountered during VCC establishment, or if the VCC is released, the primary LES/BUS will periodically (every 5 seconds) retry the call.

The backup LES/BUS will become the active LES/BUS for the ELAN if the primary LES/BUS fails. The backup LES/BUS detects the unavailability of the primary LES/BUS when the redundancy VCC is no longer present. Note, that when the primary LES/BUS becomes active again the backup LES/BUS will drop all VCCs to LE clients and will no longer accept calls. Availability of the primary LES/BUS is detected by the redundancy VCC being restored, For the redundancy protocol to be effective, LE clients must detect the failure of the primary LES/BUS and connect to the backup. LE clients detect LES failures when VCCs are released, and no new VCC can be established.

Important

LE clients will only connect to the backup LES/BUS, if they learn the LES address from the LECS. For LE clients that use a hard-coded LES address, the automatic backup mechanism described cannot be used.

Note: The IBM 8281 ATM LAN Bridge will provide LECS support at the end of 1996.

Having failed to re-establish its connections to the primary LES, the LE client issues an LE-CONFIGURE-REQUEST to its LECS. Upon receipt, the LECS determines if a local (that is, on the same MSS Server) LES is active, either as a primary or backup for the ELAN:

- If the local LES is configured as primary and active, the LECS return the ATM address of the local LES
- If the local LES is configured as backup, is active, but no redundancy VCC has been established, the LECS returns the ATM address of the local LES
- If the local LES is configured as backup and the redundancy VCC is established, the LECS will return the ATM address of the primary LES

If neither primary nor backup LES is local, the LECS maintains for each LE client a *short-memory* of the LES ATM address it has last provided to it. If an LE client sends a LE-CONFIGURE-REQUEST within the time-out period (5 minutes) for the cached entry, the LECS will return the alternate LES address and update its short-memory for this LEC. As a result, LE clients will alternate between the primary and backup LES, until the LE client makes a successful connection to an active LES.

In the normal case where the primary is active, the LEC will get the active LES address the first time it issues an LE-CONFIGURE-REQUEST. In the case of a primary LES failure, the LE client may be required to send an additional LE-CONFIGURE_REQUEST.

Note: When the primary comes back online it will restore the redundancy VCC. To avoid LE clients being connected to two different LES/BUSs in the same ELAN, the backup LES/BUS releases all its connections to LE clients.

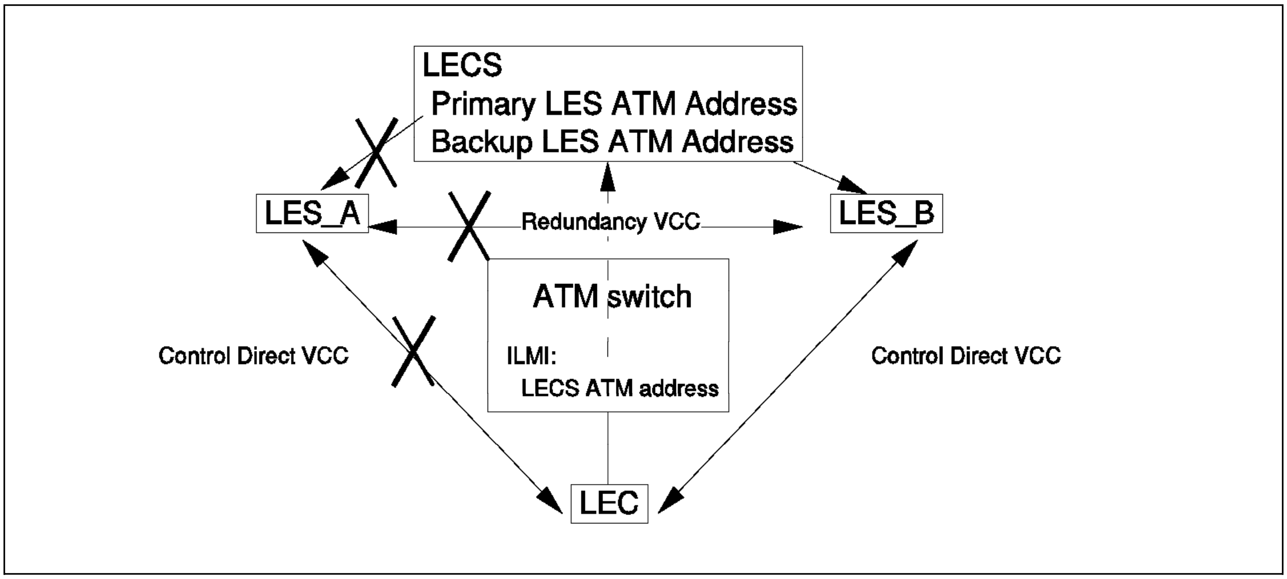


Figure 49. Failure of Primary LES/BUS

Figure 49 can be used to understand what happens is the primary LES/BUS fails. Note that the LECS and LES/BUS are server components that can run on the same or different machine.

When the primary LES/BUS fails, the control direct VCC to the LES_A will fail and the LE client has to reinitialize. During re-initialization the LE client obtains the ATM address of the LECS from the ATM switch, and connects to the LECS to learn the LES address. If either LES_A or LES_B is local to the LECS, the LECS queries it, learns which LES is active, and returns the correct ATM address. If both LES_A and LES_B are remote, LECS returns either LES_A or LES_B (depending which address has been returned to the LE client before).

5.5 Redundant LECS

The LECS is responsible for returning the active LES address to LE clients that want to join an ELAN. When an LECS becomes unavailable the LE clients that have joined the ELAN can continue to communicate, however, no new clients will be able to learn the address of the LES and connect to the ELAN.

To increase the accessibility of your ELANs you can decide to install multiple LECSs. Note, that each LECS can support multiple ELANs. Only a single LECS is supported per IBM 8210 Nways MSS Server. The ATM addresses of the LECSes must be configured in the ILMI database in all ATM switches that are adjacent to your LE clients.

Multiple LECSs will only result in increased ELAN availability if your ATM switches support the definition of multiple LECSs, and your LE clients obtain the LECS address using ILMI. Other methods, such as using the LECS well-known address, or hard-coding the LECS address in the LE client adapter, will not result in increased availability.

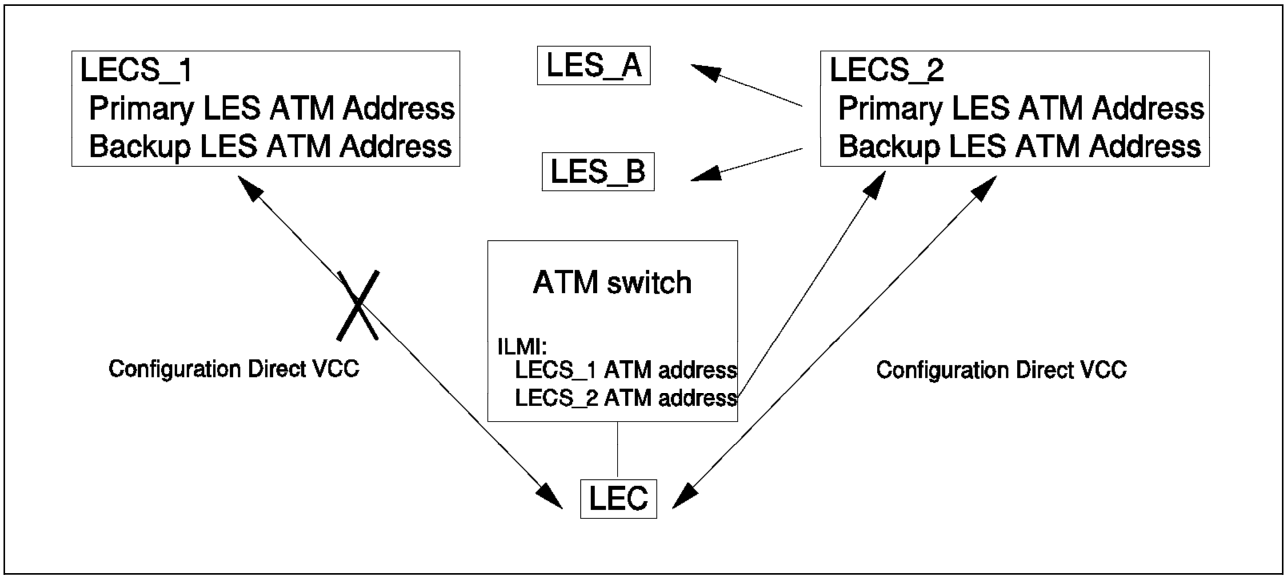


Figure 50. Failure of LECS

Multiple LECSs increase the possibility that LE clients can join an ELAN. If all your LE clients connect to the same ATM switch, the order in which LECSs are used is controlled by the order in which the LECSs have been defined in the ILMI database of the ATM switch. Note that if the LE clients are connected to multiple ATM switches, you can, by defining the LECSs in a different order on your ATM switches, have multiple LECSs active at the same time.

Recommendation

To avoid confusion, especially when you are using multiple LECSs simultaneously, make sure that when defining multiple LECSs their policies to assign LE clients to ELANs are identical.

Figure 50 can be used to understand what happens if an LECS fails. Note that the LECS and LES are server components that can run on the same or different machines.

Note: Adequate backup is provided if LECS_1 and LES_A, and LECS_2 and LES_B are on separate MSS Servers.

When an LE client wants to join an ELAN it obtains the ATM address of the first LECS in the switch′s database, and attempts to establish a configuration direct VCC. When this VCC fails, the LE client obtains the ATM address of the second LECS in the ILMI database and retries the VCC. When VCC establishment is successful, the LE client sends a CONFIGURE_REQUEST and learns the ATM address of the active LES.

For configuration details, see 5.11, “Configuring LAN Emulation Services” on page 120.

5.6 MSS Server Policies

One of the benefits of ATM is that it allows for the creation of emulated LANs based on logical characteristics of the clients, independent of where a client is physically located. The IBM 8210 Nways MSS Server supports a flexible and efficient method to assign LE clients to ELANs based on the information exchanged during the LAN emulation configuration server connect phase, and ELAN policies defined on the MSS Server.

During the configuration phase the LE client learns the ELAN configuration information, including the LES address, of the ELAN to which the LECS assigns the LE client. The MSS Server returns these values based on its ELAN and its policy definitions.

The MSS Server′s ELAN policies can be based on:

1. ATM address prefix

   A policy based on ATM address helps the LECS to return an LES address based on the complete, or part of, the LE client′s ATM address

2. ELAN name

   An ELAN name policy directs the LECS to return a LES address based on the ELAN name the LE client has included in its LE_CONFIGURE_REQUEST.

3. ELAN type

   A policy based on ELAN type (that is, Ethernet or token-ring) helps the LECS to return a LES address based on the LAN type identifier the LE client has included in its LE_CONFIGURE_REQUEST.

4. MAC address

   A policy based on MAC address helps the LECS to return a LES address based on the unicast MAC address the LE client has included in its LE_CONFIGURE_REQUEST.

5. Maximum frame size

   A policy based on maximum frame size helps the LECS to return a LES address based on the maximum frame size the LE client has included in its LE_CONFIGURE_REQUEST.

6. Route descriptor (source route bridging proxies only)

   A policy based on route descriptor helps the LECS to return a LES address based on the route descriptor the LE client has included in its LE_CONFIGURE_REQUEST.

   Note: The route descriptor is an optional field in the LE_CONFIGURE_REQUEST, that will only be included by SRB proxies.

Note: The policies correspond with fields which may be present in the LE_CONFIGURE_REQUEST sent by the LE client to the LECS. For details see “Configuration Phase” on page 79.

5.6.1 LECS Assignment Policy Details

To understand how the MSS Server applies the policies defined, it is important to differentiate between *policy types*, and *policy values*. Policy types are associated with the LECS, while the policy values are associated with ELANs. Both need to be defined on the MSS Server on which the LECS resides.

Per LECS you specify which policy types, one or more of the six types specified in the previous section, are enabled and the priority that is associated with it. For example Figure 51 indicates a MSS Server on which the ATM address prefix, the ELAN name, and the ELAN type policies have been enabled, using priority 10, 20, and 30 respectively.

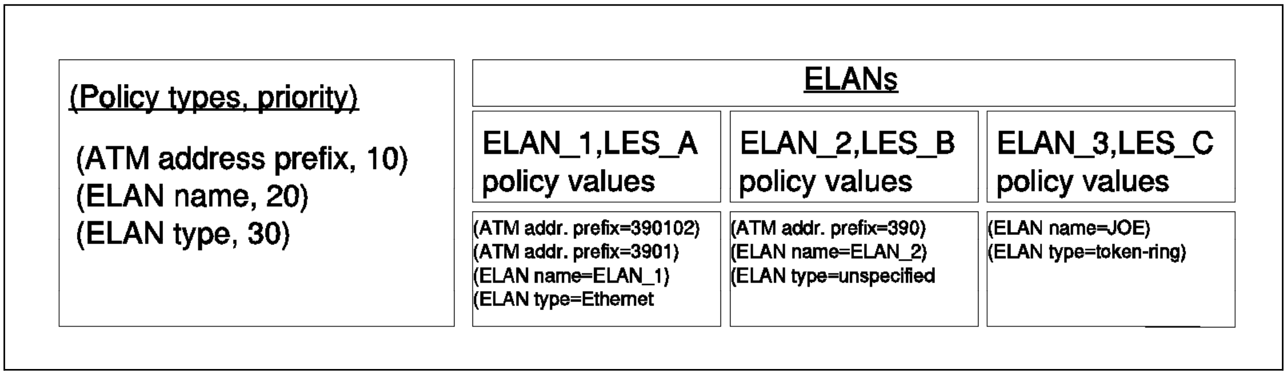


Figure 51. ELAN Policies

Note: Policy types that have the same priority will be AND′ed.

In addition to enabling and prioritizing the priority types, you must define one or multiple policy values for each of your ELANs. There is no need to define a policy value for each enabled type. In Figure 51, for example, we have two ATM address prefix policy values (390102 and 3901 respectively) defined for ELAN_1, one ATM address prefix policy value for ELAN_2, but none for ELAN_3.

Important

LE clients can only be assigned to an ELAN if you have defined at least one policy value for the ELAN. If you do not assign policy values to an ELAN, no client will ever be assigned to it. Make sure the policy type is enabled and a priority has been assigned.

Enabling a policy type means that the corresponding policy values can be applied. Note, that it does not make sense to define policy values for disabled policy types. The LECS will never apply these policies. The priority (small number means high priority) indicates the order in which policies values of different type are interpreted. For example enabling the ATM address prefix policy with priority 10 and the ELAN name policy with priority 20, means that selecting on an ELAN based on ATM address prefix will be tried before selecting an ELAN based on ELAN name.

ELAN policy values can be configured in any order, however, the LECS interprets the policy values in the order of priorities assigned to its policy types. In the example, if an LE client registers than the LECS performs a policy check in the following order:

1. If the ATM address prefix policy value applies

2. If the ELAN name policy value applies
3. If the LAN type policy value applies

To make sure the LECS assigns LE clients to the proper ELAN you have to make sure that policy values are non-ambiguous. For this purpose the Configuration Program enforces uniqueness of:

- ATM prefix policy values
- ELAN name policy values
- MAC address policy values
- Route descriptor policy values

For example, defining an ATM address prefix policy value equal to 3901 for multiple ELANs is not allowed as the LECS is unable to decide to which ELAN an LE client should be assigned when this policy applies.

Note: The Configuration Program does not enforce uniqueness of the ELAN type and maximum frame size policy values.

If none of the policies applies the LE_CONFIGURE_REQUEST will fail. Therefore, although it is no guarantee that the policy is successful for all LE clients, you need to define at least one policy value per ELAN.

5.6.1.1 Default Policies

LE clients will not be assigned to an ELAN, unless there is at least one ELAN for which a policy value has been defined that matches the LE client′s configure request. Due to the fact that many of the LE_CONFIGURE_REQUEST fields are optional it makes sense to define a *default* policy. A default policy is a set of policy values that always apply and enables ELAN assignment when none of the other policy value apply. Make sure that the policy type has the lowest priority.

In the following sections we describe each of the policy types in more details. Item 8 on page 132 discusses how to configure ELAN policy values using the Configuration Program.

5.6.1.2 ATM Address Prefix Policy

An ATM address policy directs the LECS to return a LES address based on part of, or the whole, LE client′s ATM address included in its LE_CONFIGURE_REQUEST.

The MSS Server LECS permits variable-length ATM address prefixes. For example, the policy value (39.01.02, LES_A), implies that ATM addresses beginning with the above prefix, should be assigned to LES_A. Using a given LE client′s ATM address, the LECS searches for the ATM address policy value with the longest matching prefix. Thus, for example, the above policy value would take precedence over policy value (39, LES_B).

The ATM address is the only policy value that indicates any geographic information as the network prefix portion of the ATM address identifies the ATM switch to which the LE client is attached. This policy should be used when geographic-based assignment is needed. For example, this policy can be used to identify all LE clients belonging to the same switch, and assign them to the same ELAN.

The ATM address policy should also be used when security is a major concern. The highest level of security is achieved when this policy is used in conjunction with the LES/LECS security extension (see 5.7, "Secure ELANs" on page 115).

5.6.1.3 ELAN Name Policy

An ELAN name policy directs the LECS to return a LES address based on the ELAN name the LE client has included in its LE_CONFIGURE_REQUEST. LE clients have the option to refer to the actual name or use an alias.

When LE clients refer to the ELAN using its real ELAN name, for example ELAN_1 serviced by LES_A, make sure that you create policy (ELAN_1, LES_A), which means that LE clients that specify ELAN_1 in their configuration requests will be assigned to LES_A.

The use of aliases allows LE clients to refer to an ELAN by an alias rather than the real ELAN name (as used by the LES). For example, all LE clients belonging to the accounting department could be configured to use the ELAN name ACCOUNTS, while members of the engineering department could use ELAN name ENGINEERING. The following LECS configuration policies could then be configured on the MSS Server:

```
(ACCOUNTS: LES_A)
(ENGINEERING: LES_B)
```

Alternatively, if the number of clients is small, you may want to include them all in a single ELAN. In that case you could use the following policy definitions in the LECS:

```
(ACCOUNTS: LES_A)
(ENGINEERING: LES_A)
```

Note that policies based on the ELAN name are case sensitive.

A special example of using aliases is to allow each LE client to use its own alias. For example, you could create the policy values (JOE, LES_A) and (MARY, LES_A). The LE clients configured with those names will be assigned to the same LES (and hence the same ELAN).

Advantage of using aliases per LE client is that you can, by reconfiguring the policy for a specific name, easily move LE clients from one ELAN to another. Disadvantage is that the number of aliases that you need to define on the MSS Server can become very large, which can turn your configuration into a administrative nightmare.

Important

Note: The maximum number of policies is limited (the maximum is currently 1500 per LECS). Make sure you do not exceed this number.

ELAN name policies are the simplest and most flexible type of policy to implement. Make sure that when you have multiple ELANs that you have a consistent naming convention, and that your LE clients station are configured accordingly.

5.6.1.4 ELAN Type Policy

An ELAN type policy directs the LECS to return a LES address based on the ELAN type included in the LE_CONFIGURE_REQUEST. Valid values are:

- Unspecified
- Ethernet
- Token-ring

Note: Unspecified applies when no ELAN type has been included in the register request.

ELAN type policy values are most useful for assigning a default ELAN. For example, the following policy values would ensure that every LE client is assigned to one of the LESes:

```
(ELAN Type = token-ring: LES_A)
(ELAN Type = Ethernet: LES_B)
(ELAN Type = unspecified: LES_C)
```

Note: Give the policies used for providing default ELAN assignments a low priority, so that the more specific policies are applied first.

5.6.1.5 MAC Address Policy

A MAC address policy directs the LECS to return a LES address based on the 6-byte LE client′s MAC address.

Note: Do not confuse the 6-byte ESI field within the ATM address with the MAC address. LE clients may use the ESI also as their MAC address, but might as well use a different value.

This policy type can be used to ensure that an LE client is assigned to the proper ELAN, regardless of its physical location in the network. For example, when a workstation is being moved from one location to another but wants to retain its membership in the same ELAN.

When using a MAC address policy, each of your MAC address policy values must be unique. Make sure you enter a valid MAC address.

5.6.1.6 Maximum Frame Size

A maximum frame size policy directs the LECS to return a LES address based on the maximum frame size type included in the LE_CONFIGURE_REQUEST. Valid values are:

- Unspecified
- 1566
- 4544
- 9234
- 18190

Unspecified applies when no maximum frame size has been included in the register request.

The maximum frame size policy can be used to assign a default ELAN. For example, the following policy values would ensure that every LE client is assigned to one of the LESs:

```
(Maximum frame size = 1516: LES_A)
(Maximum frame size = 4544: LES_B)
(Maximum frame size = 9234: LES_B)
(Maximum frame size = 18190: LES_B)
(Maximum frame size = unspecified: LES_C)
```

Note: Give the policies used for providing default ELAN assignments a low priority, so that the more specific policies are applied first.

5.6.1.7 Route Descriptor Policy

A route descriptor policy directs the LECS to return a LES address based on the 2-byte LE client′s route descriptor. Be aware that the route descriptor is only used by proxy stations that perform source route bridging (SRB).

When using a route descriptor policy, each of your route descriptor policy values must be unique.

5.7 Secure ELANs

If the ATM address of the LAN emulation server (LES) is known, it is possible for an LE client to join the ELAN supported by that LES, simply by configuring the ATM address of the LES. A limitation of the current version of the ATM Forum′s LAN Emulation specification (Version 1.0) is the absence of any mechanism to prevent clients bypassing the LECS, and directly joining an ELAN.

To overcome this limitation the MSS Server provides a security feature to control ELAN membership. The MSS LECS/LES security feature consists of two parts:

1. A secure LES will validate LE clients with the LECS, before allowing them to join the ELAN
2. An LECS that supports validates request from the secure LES

Note: The secure LES and the LECS can reside on different MSS Servers.

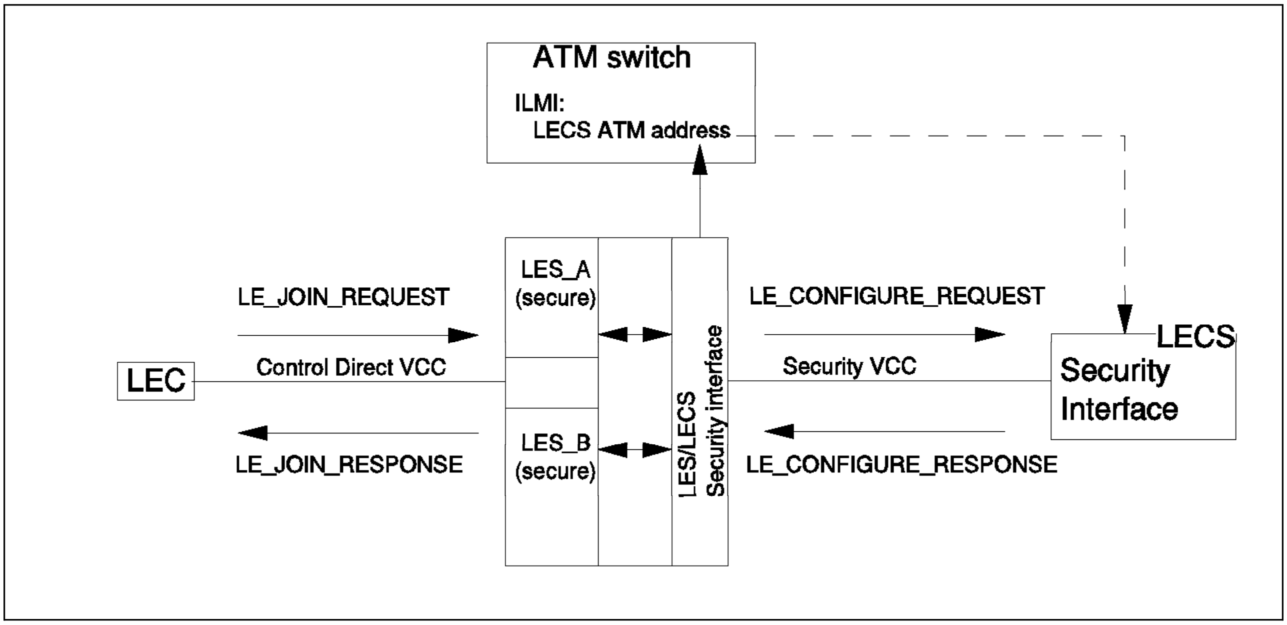


Figure 52. Secure ELANs

Before accepting LE clients, the LES will trigger its LES/LECS security interface. The LES/LECS security interface sends an LE_CONFIGURE_REQUEST on behalf of the client to the LECS, using the information in the client′s LE_JOIN_REQUEST received on the control direct VCC. To reduce the number of VCCs required between the LES/LECS security interface and the LECS the LE_REGISTER requests and responses exchanged are multiplexed on a single security VCC.

Notes:

1. A LES/LECS security interface is associated with an ATM port and will service all LES′es that are associated with the same port.
2. All clients join requests will be verified, including the requests coming from (well behaving) clients that have learned the LES address from the LECS.

To enable the LES/LECS security interface to establish the security VCC, it learns the LECS address from the adjacent switch in the same way as LE clients learn the LECS address (see 4.4.3.1, "Initialization" on page 77).

Note: If LES and LECS functions are residing on the same MSS server an internal interface is used instead of the security VCC. ILMI, however, is required to learn that the LECS and LES are local.

The variables the LE client has included in the JOIN-REQUEST are copied in the LE_CONFIGURE_REQUEST. In addition the ATM address and the ELAN name of the requesting LES are included in an IBM security TLV. As the LECS returns this TLV unchanged in the LE_CONFIGURE_RESPONSE it is possible to easily identify the LES that requested client verification.

To maximize the security of an ELAN, the following steps are recommended:

- Use ATM address policy at the LECS to assign ELAN membership
- Activate the LECS interface(s)
- Activate the security option of the LES
- Use address screening at the ATM switches (which cause switches to validate that calling stations use their actual ATM address in call setups; thus, preventing stations from impersonating other stations)

For details on how to configure a secure LES for a particular ELAN and enabling the LECS security option, see 5.11, "Configuring LAN Emulation Services" on page 120.

5.8 BUS Monitor

The broadcast and unknown server (BUS) is a resource which is shared by all members of an ELAN, and a potential source of congestion in the network. LE clients wishing to send data to destinations whose ATM addresses have not been resolved yet by the LES will send their unicast data frames to the BUS to be broadcast to all clients in the ELAN. It is possible that some LE Clients could over-utilize the BUS to handle their data frames, as opposed to setting up a data direct VCC to the destination client, once the address resolution has been done by the LES.

The IBM 8210 Nways MSS Server provides a bus monitor function which allows network administrators to monitor the usage of the BUS by LE clients and downstream stations. When enabled, it periodically samples the traffic sent to the BUS on a particular ELAN. At the end of each sample interval, the BUS monitor identifies the top users of the BUS by their source MAC address, LEC ATM address, and the number of frames each of them sent to the BUS. Configurable parameters for this option are:

- Number of top users to record
- Number of seconds in each sample interval
- Sample rate (that is, sample one out of every “sample rate” frames
- Number of minutes between sample intervals

The statistics collected by bus monitor for a particular ELAN can be displayed using the command line interface. For example:

```
*talk 5
+network 0
ATM Console
ATM+le-services
LE-SERVICES+work ET1
EXISTING LES-BUS ′ET1′+
EXISTING LES-BUS ′ET1′+statistics display bus-monitor

-BUS Monitor Status-
Currently in a sample interval ?            no
Next sample interval scheduled in:          0 minute(s), 19 second(s)

-Results of Last Complete Sample-
BUS Monitor sample interval started at:     00.01.05.13   (System UpTime)
Duration of sample interval:                10 second(s)
# Top Hosts Actually Recorded:              2
# Frames Received in sample interval:       16
# Frames Sampled in sample interval:        2
Frame sampling rate:                        1 out of 10
                                                                     # frames
Rank  Source MAC Addr.    Associated LEC ATM Address                 sampled
----  -----------------   ----------------------------------------   ----------
   1  02.00.4A.00.4A.6A   3909851111111111111111010102004A004A6A81          1
   2  5A.00.00.00.00.00   3909851111111111111111010160008210000004          1
EXISTING LES-BUS ′ET1′+
```

5.9 Type/Length/Values (TLVs)

Type/Length/Values (TLVs) enable network administrators to exercise control over certain LE client characteristics. TLVs can be defined during the definition of your ELANs and are specified on an ELAN basis.

Upon receipt of an LE_REGISTER_RESPONSE from the LECS, LE clients must alter the local TLVs according to the ones set in the TLV entries received. This enables you to administer and enforce specific LE clients characteristics from a central location. For example, when LE client cells travel over many ATM switches before reaching the LES, the default timeout may be insufficient. Defining an appropriate TLV may circumvent this.

The TLVs that can be encoded are listed below. The numbers conform to the ATM Forum LAN Emulation specification Version 1.0. For an explanation of these parameters, see Appendix A, “ATM Forum-Compliant Frame Formats” on page 465.

C7 Control time-out

C10	Maximum unknown frame count
C11	Maximum unknown frame time
C12	VCC time-out period
C13	Maximum Retry Count
C17	Aging time
C18	Forward delay time
C20	Expected LE_ARP response time
C21	Flush time-out
C22	Path switching delay
C23	Local segment ID
C24	Multicast Send VCC type
C25	Multicast send VCC average rate
C26	Multicast Send VCC peak rate
C28	Connection completion timer

The TLVs that have been defined for a particular ELAN can be displayed using the command line interface.

5.10 Redundant Spanning Tree Root Bridge

Figure 53 on page 119 depicts a scenario where MSS Server A has two functions:

- LES/BUS function to establish an ELAN

  The LES/BUS function is used to establish an ELAN which is used as a (fast) backbone LAN segment for providing connectivity between legacy LANs. The LES/BUS is defined as the primary LES/BUS for the ELAN. Note that the traffic between the LE clients on the legacy LAN bridges is direct, that is, it does not use the bridging functions on the MSS Server.

- LE client function

  An LE client has been defined which is configured as a transparent bridge. By setting the bridge priority the highest in the network this TB will become root of the spanning tree. Therefore it will periodically send BPDUs (bridge PDUs).

To provide backup if MSS Server A fails, equivalent functions have been defined on MSS Server B:

- Backup LES/BUS function to establish an ELAN

  The LES/BUS function is defined as a backup for the primary LES/BUS on server A. It will take over responsibility for the ELAN when A fails.

- Backup client function

  A backup LE client has been defined that is equivalent to the LE client on A. However, it has hard-coded the local (backup) LES/BUS as its LES/BUS for the ELAN. Because the backup LES/BUS will only accept LE clients when the primary becomes unavailable (see 5.4, “Redundant LES/BUS” on page 106), the LE client on B will join the ELAN only after MSS Server A fails.

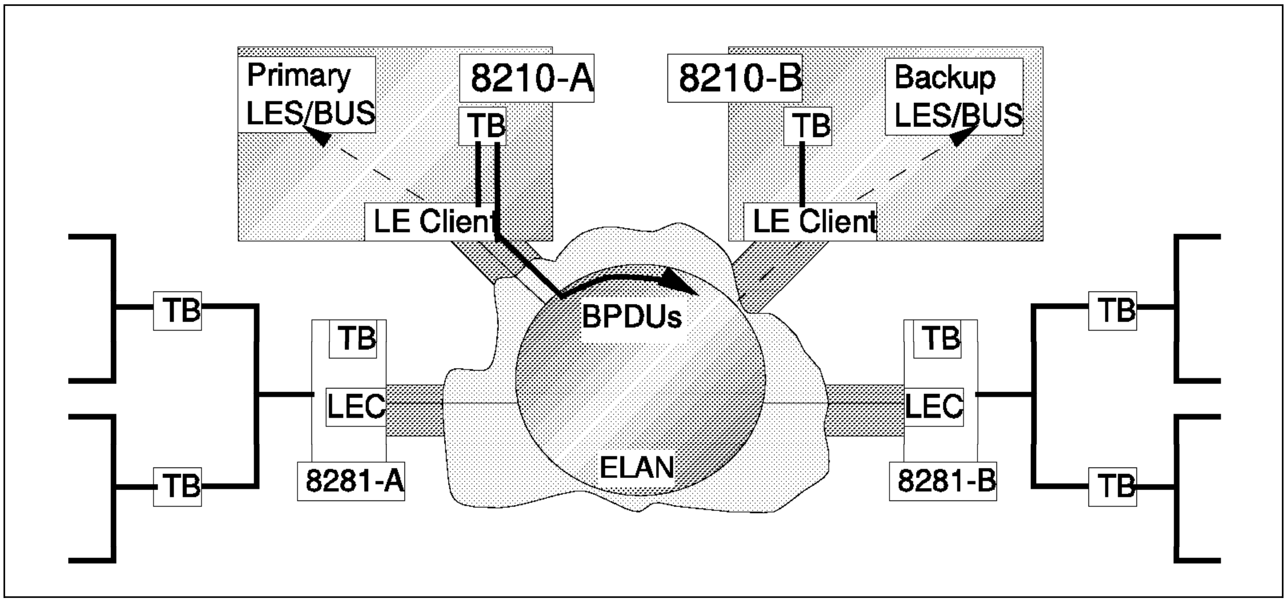


Figure 53. Redundant Spanning Tree Root Bridge

As a result, server B will take over the LES/BUS and the root bridge function when server A fails. If server A becomes active again, it will regain LES/BUS and root bridge function.

Important to realize in this backup mechanism is:

1. The primary TB LE client will take over when the primary LES becomes available again.
2. Although the LE client on the MSS Server is configured as a transparent bridge, no actual bridging takes places. The LE client only participates in the maintenance of the spanning tree.
3. Because of its reliability and availability, due to, for example, the dual power supply on the 8260, the 8260 MSS Server blade is an excellent choice for root bridge, even if not using the redundancy feature.
4. It is essential that all TB LE clients attached to the ELAN, learn the LES/BUS address from the LECS. LECS support for the IBM 8281 ATM LAN Bridge becomes available at the end of ′96.
5. The take-over from primary to backup server should not impact the spanning tree status as maintained on the ATM-attached legacy LAN bridges. Adequate timer settings on the bridges is required.

In the scenario depicted there is no need to attach both 8210s to the same ATM switch, as the ATM addresses of the LES/BUSs and the TB LE clients can be different. It requires however that all LE clients use an LECS to learn the LES address.

If you have LAN-ATM bridges that do not support an LECS an alternative method to provide the same level of backup is:

- Connect both MSS Servers to the same ATM switch
- Configure an identical locally administered ESI on A and B
- Configure identical LES/BUS and LE client on A and B using the locally administered ESI

- Define both LES/BUS as primary LES/BUS without backup
- Hard-code LES address on LAN-ATM bridges

Because the ATM switch to which the MSS Servers attach will not allow registration of the same ESI from both servers, only one of the server′s LES/BUS and LE client functions will become active. The backup server takes over when the primary fails. The backup server retries ESI registration every 30 seconds. Its ESI registration will be successful when the primary server′s ESI is de-registered.

Note: An example which depicts 8210 and 8281 configuration for this scenario is detailed in 11.4.23, “Spanning Tree Root Bridge Redundancy” on page 445.

5.11 Configuring LAN Emulation Services

This section covers the basic steps required to configure MSS Server′s LAN emulation functions using the Configuration Program.

Note: Definition scenarios using the command line interface are included within Chapter 11, “Implementation Scenarios” on page 289.

The different configuration steps that need to be completed depend on the functions that need to be activated. The basic steps include:

1 Define the ATM port(s)

LE clients, LES/BUS, LECS, and LECS/LES security feature are components of the 8210 that need to be associated with an ATM port. When associated, all ATM traffic of the component will take place on the ATM port specified.

During configuration of the ATM port you have to configure:

a. The ATM port attributes

For configuration details see Chapter 3, “MSS Server and ATM Ports” on page 53.

b. The end system identifiers (ESIs) associated with the ATM port

During the definition of any of your LE server and client components either the burned-in end system identifier (ESI) or a user-defined ESI must be used. To ease the definition of ATM addresses on remote LE clients and ATM switches, and to simplify troubleshooting, it is recommended that you use a locally administered ESI for the LECS and LES/BUS services. ESIs are administered per ATM port. A separate value has to be defined per ATM port. All LE client/servers that connect using a particular port can use the same ESI. Make sure that when defining multiple 8210s, the user-defined ESIs are unique.

The ATM port configuration must be repeated for each ATM port on your MSS Server which is used for LE functions.

2 Configure your ELAN(s)

This step needs to be repeated for every ELAN for which the 8210 performs LES/BUS and/or LECS functions. Configuration information that must be entered is:

a. ELAN name and type

Define the ELAN name and type (token-ring or Ethernet). Indicate if the MSS Server is performing LES/BUS functions for this ELAN. If yes, associate the LES/BUS with an ATM port and assign an ESI and SEL.

b. ELAN security

Define the ELAN as a secure ELAN (see 5.7, "Secure ELANs" on page 115) to prevent LE clients from bypassing the LECS before joining the ELAN.

Note: This step should only be considered if the MSS Server is performing LES/BUS functions for this ELAN. Do not forget to activate the LECS/LES security feature in addition.

c. Intelligent LES/BUS

Indicate if the intelligent LES and/or intelligent BUS functions are being used for this ELAN. For details on ILES and IBUS see 5.1.1, "Intelligent LES (ILES)" on page 89 and 5.1.2, "Intelligent BUS (IBUS)" on page 89, respectively.

Note: This step is only required if the MSS Server is performing LES/BUS functions for this ELAN.

d. BUS monitoring

Indicate if the BUS monitoring function is being used for this ELAN. For details see 5.8, "BUS Monitor" on page 116.

Note: This step should only be considered if the MSS Server is performing LES/BUS functions for this ELAN.

e. Broadcast manager (BCM) and source route manager (SRM)

Indicate if/how BCM and SRM functions are being used for this ELAN. For details see 5.2, "Broadcast Manager (BCM)" on page 90 and 5.3, "Source Route Manager (SRM)" on page 103, respectively.

Note: This step should only be considered if the MSS Server is performing LES/BUS functions for this ELAN.

f. BCM static definitions

Define static BCM definitions for IPX stations. For details see 5.2.4.1, "Static IPX/BCM Definitions - Passive Device" on page 97.

Note: This step should only be considered if the MSS Server has enabled BCM functions for IPX traffic.

g. LES/BUS redundancy

Indicate if LES/BUS redundancy is being used. Define the LES/BUS as primary or backup. For details see 5.4, "Redundant LES/BUS" on page 106.

Note: This step should only be considered if the MSS Server is performing LES/BUS functions for this ELAN.

h. ELAN policy values

Indicate the policy values for this ELAN, which enable the local LECS to assign LE clients to this ELAN. For details see 5.6, "MSS Server Policies" on page 110.

Note: Policy types and values need only to be defined when the LECS function has been activated on the MSS Server.

i. Define TLVs for this ELAN

   Indicate which TLVs are returned by the LECS when an LE client connects to an ELAN. For details see 5.9, “Type/Length/Values (TLVs)” on page 117.

   Notes:

   1) This step should only be considered if the MSS Server is performing LECS functions for this ELAN.

   2) Consider adding user-defined TLVs or modifying the system-defined TLVs (5.11.4, “Defining and Modifying TLVs” on page 140), when the configurator supplied TLVs are not adequate.

The previous steps need to be repeated for every ELAN that needs to be configured on your MSS Server. For details see 5.11.1, “Configuring ELANs” on page 123.

3 Configure LECS

This step needs to be done if the 8210 performs LECS functions. Information that needs to be entered is:

a. Create and enable LECS instance

   If the MSS Server is performing LECS functions, associate the LECS with an ATM port and assign an ESI and SEL.

b. Define LECS policies/priority

   Define the policy types and their priority that are used by the LECS to assign LE clients to ELANs. For details see 5.6, “MSS Server Policies” on page 110.

   Note: The definition of LECS policy types need to be done in conjunction with the policy values defined during ELAN definition.

As each 8210 can have a single LECS instance only, the previous steps need to be done only once. For configuration details see 5.11.2, “Configuring LECS” on page 133.

4 Define TLVs

During the definition of your ELANs you can specify which TLVs are returned by the LECS to its LE clients. You can select user-defined or system-defined TLVs.

Configuration options exist to modify the system-defined TLVs, or add your own user-defined TLVs. For details see 5.11.4, “Defining and Modifying TLVs” on page 140.

This step is optional and should only be considered if your MSS Server performs LECS functions, TLVs are used, and the system-defined TLVs are not adequate.

5 Configure LECS/LES security interface

The LECS/LES security feature needs to be enabled if you have defined secure ELANs. During its definition associate the security interface with an ATM port, and assign an ESI and SEL to it.

Note: This step is required in addition to configuring secure LESs during ELAN definition.

Defining the security interface is mandatory if you have defined at least one secure ELAN. For details see 5.11.3, “Configuring LECS/LES Security Interface” on page 138.

6 Configure LE clients

Basic definition steps to define your LE clients include:

a. Define LE client addresses

   Associate the LE client with an ATM port and specify its ESI, SEL, and MAC address. Define the LE client to be compliant with either IBM LAN emulation or ATM Forum LAN emulation.

b. Define the ELAN name and type

   Specify the ELAN name, type and maximum frame size.

c. Define LE server (LECS or LES)

   Identify how the LECS or LES ATM address is obtained (hard-coded or dynamically).

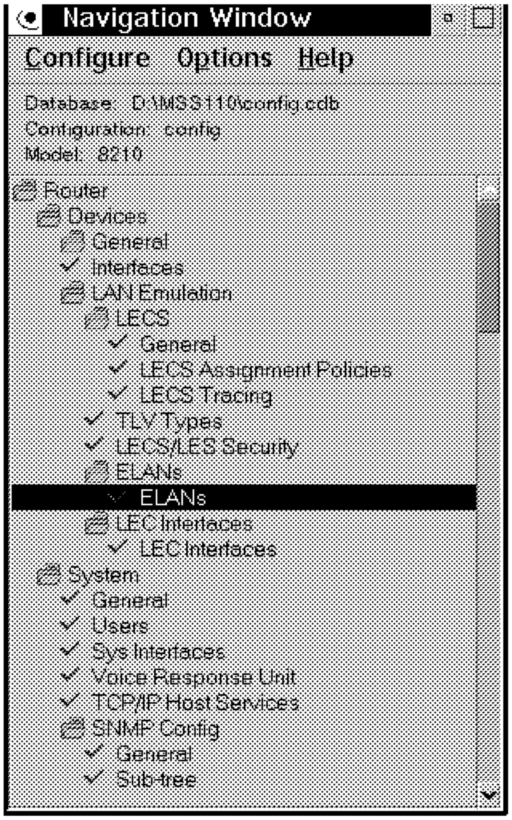
d. Define higher-layer functions

   In addition to the basic LE client configuration, higher-layer (bridging or routing) functions need to be configured. For details on IP routing, IPX routing, and bridging, see Chapter 7, “MSS Server and IP Routing Protocols” on page 169, Chapter 8, “MSS Server and IPX Routing” on page 205, and Chapter 9, “MSS Server and Bridging” on page 233.

The previous steps need to be repeated for every LE client. For configuration details see 5.11.5, “Configuring LE Clients” on page 142.

5.11.1 Configuring ELANs

Configuring an ELAN can be done by clicking on ELANs in the Navigation Window (see Figure 54 on page 124).



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

- Router
  - Devices
    - General
    - ✓ Interfaces
    - LAN Emulation
      - LECS
        - ✓ General
        - ✓ LECS Assignment Policies
        - ✓ LECS Tracing
      - ✓ TLV Types
      - ✓ LECS/LES Security
      - ELANs
        - ✓ ELANs
      - LEC Interfaces
        - ✓ LEC Interfaces
  - System
    - ✓ General
    - ✓ Users
    - ✓ Sys Interfaces
    - ✓ Voice Response Unit
    - ✓ TCP/IP Host Services
    - SNMP Config
      - ✓ General
      - ✓ Sub-tree

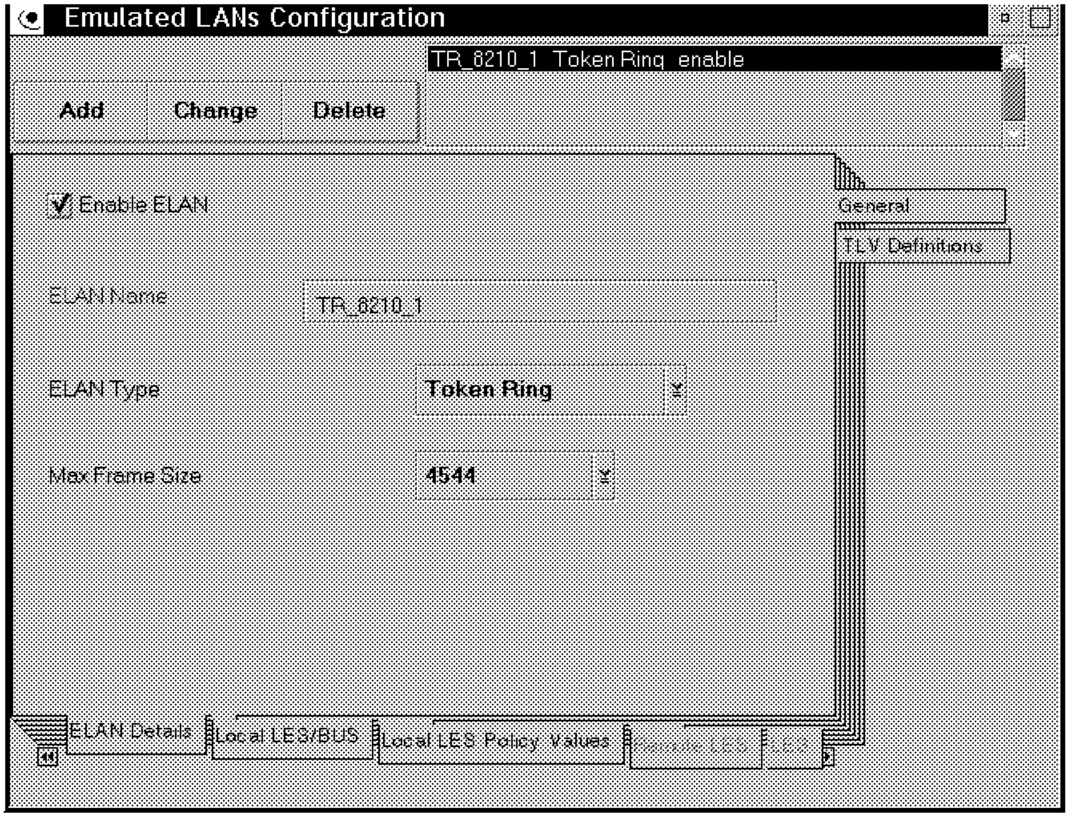
Figure 54. Configuring an ELAN

Note: ELAN configuration is required for each ELAN for which the MSS Server performs LECS and/or LES/BUS functions. The type of configuration depends on the type of LECS and/or LES/BUS functions performed.

The configuration steps required are:

1 Define the ELAN name and type

After ELANs has been selected, Figure 55 on page 125 appears. The mandatory steps are to specify the ELAN name, ELAN type (token-ring or Ethernet) and maximum frame size.



TR_8210_1 Token Ring enable

☑ Enable ELAN

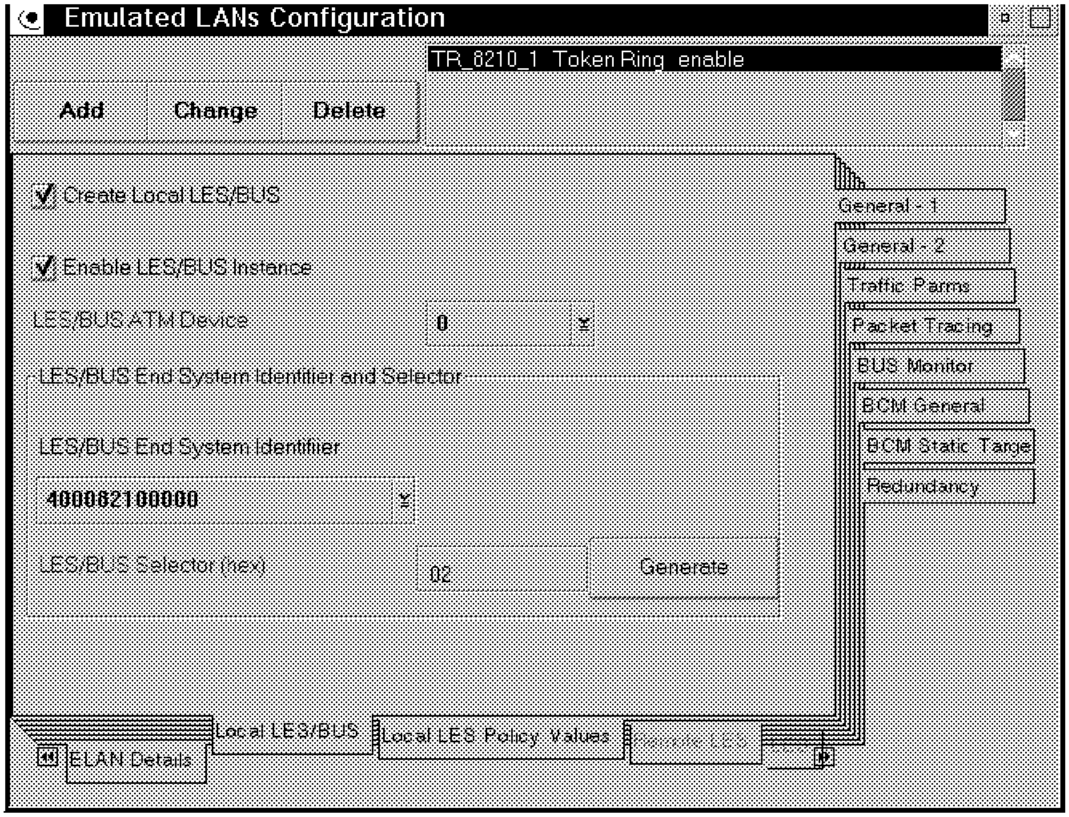
ELAN Name TR_8210_1

ELAN Type **Token Ring**

Max Frame Size **4544**

Figure 55. Emulated LAN Configuration - ELAN Details

Selecting Local LES/BUS during ELAN configuration enables you to specify the local, if any, LES/BUS attributes. As a result, Figure 56 on page 126 appears.



TR_8210_1 Token Ring enable

☑ Create Local LES/BUS

☑ Enable LES/BUS Instance

LES/BUS ATM Device: 0

LES/BUS End System Identifier and Selector

LES/BUS End System Identifier: 400082100000

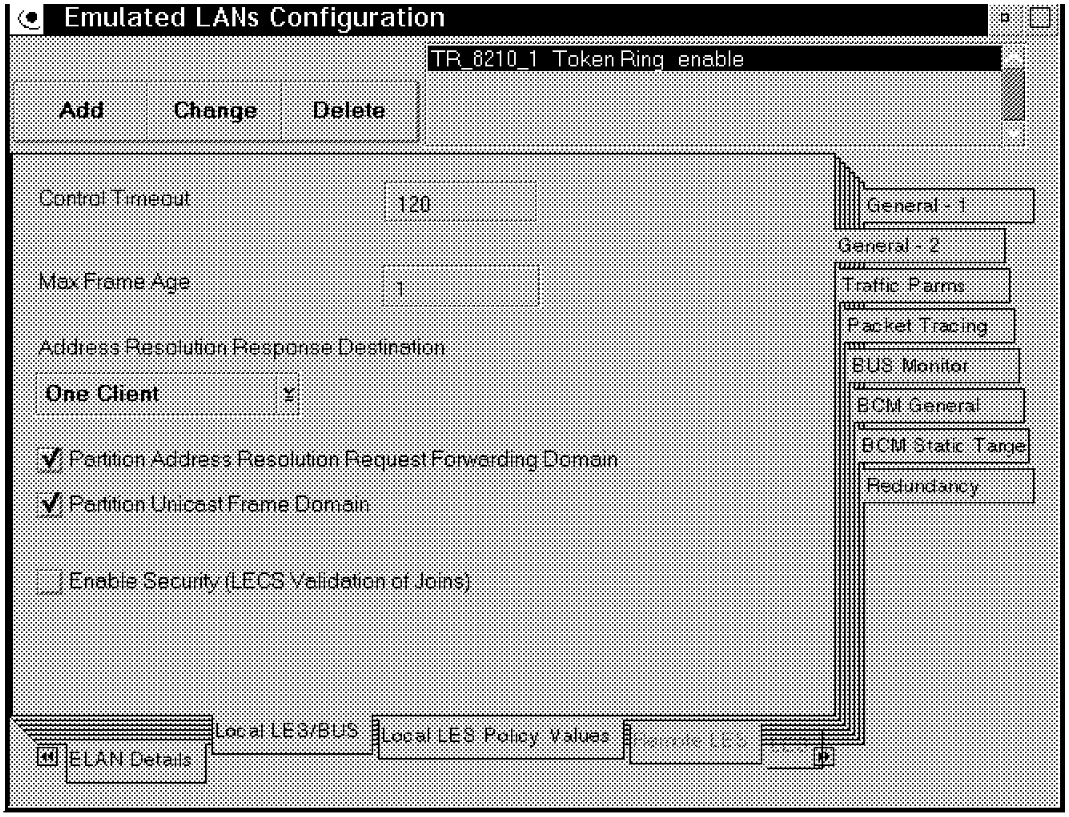
LES/BUS Selector (hex): 02

Figure 56. Emulated LAN Configuration - Local LES/BUS

Indicate if the MSS Server is performing local LES/BUS functions for this ELAN. If so, enable it, associate the LES/BUS with one of the ATM ports, and select the ESI and SEL used for constructing the ATM address used by this ELAN′s LES and BUS. It is advised that you use a locally administered ESI. To prevent the SEL byte from changing in future configurations, consider a manually generated SEL.

2 ELAN security

Selecting General-2 during local LES/BUS ELAN configuration results in Figure 57 appearing.



TR_8210_1 Token Ring enable

Control Timeout 120

Max Frame Age 1

Address Resolution Response Destination

One Client

☑ Partition Address Resolution Request Forwarding Domain

☑ Partition Unicast Frame Domain

□ Enable Security (LECS Validation of Joins)

Figure 57. Emulated LAN Configuration - Local LES/BUS. Activate ILES, IBUS and LECS/LES security.

A configuration option exists to broadcast an LE_ARP response to all LE clients or to the LE client that issued the LE_ARP request. By default, only the latter will receive the response.

Enabling security (LECS Validation of Joins) activates the LES/LECS security feature.

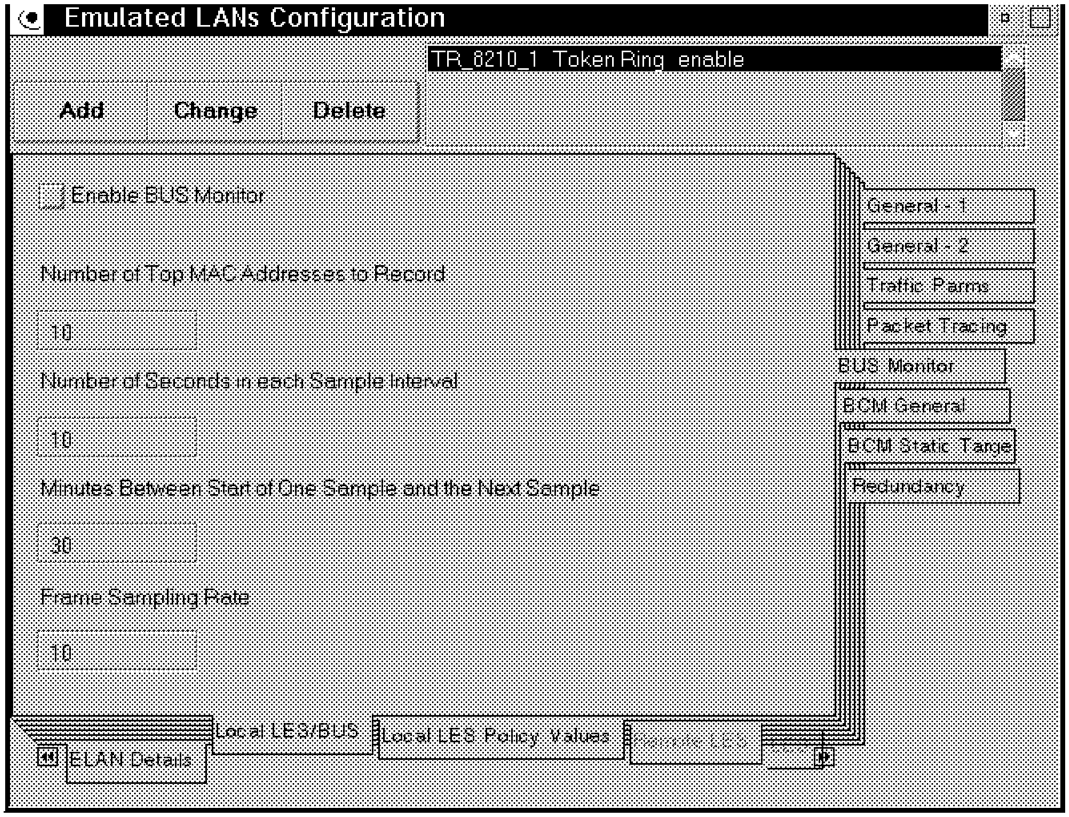
Note: If defining secure ELANs, do not forget to enable the LECS/LES security interface (see 5.11.3, “Configuring LECS/LES Security Interface” on page 138).

3 Intelligent LES/BUS

Selecting Partition Address Resolution Request Forwarding Domain in Figure 57 activates the ILES function. Select Partition Unicast Frame Domain to enable the IBUS function.

4 BUS monitoring

Selecting BUS Monitor during Local LES/BUS configuration results in Figure 58 appearing. This screen allows you to enable and configure the BUS monitoring function for this ELAN.



Emulated LANs Configuration

TR_8210_1 Token Ring enable

□ Enable BUS Monitor

Number of Top MAC Addresses to Record

10

Number of Seconds in each Sample Interval

10

Minutes Between Start of One Sample and the Next Sample

30

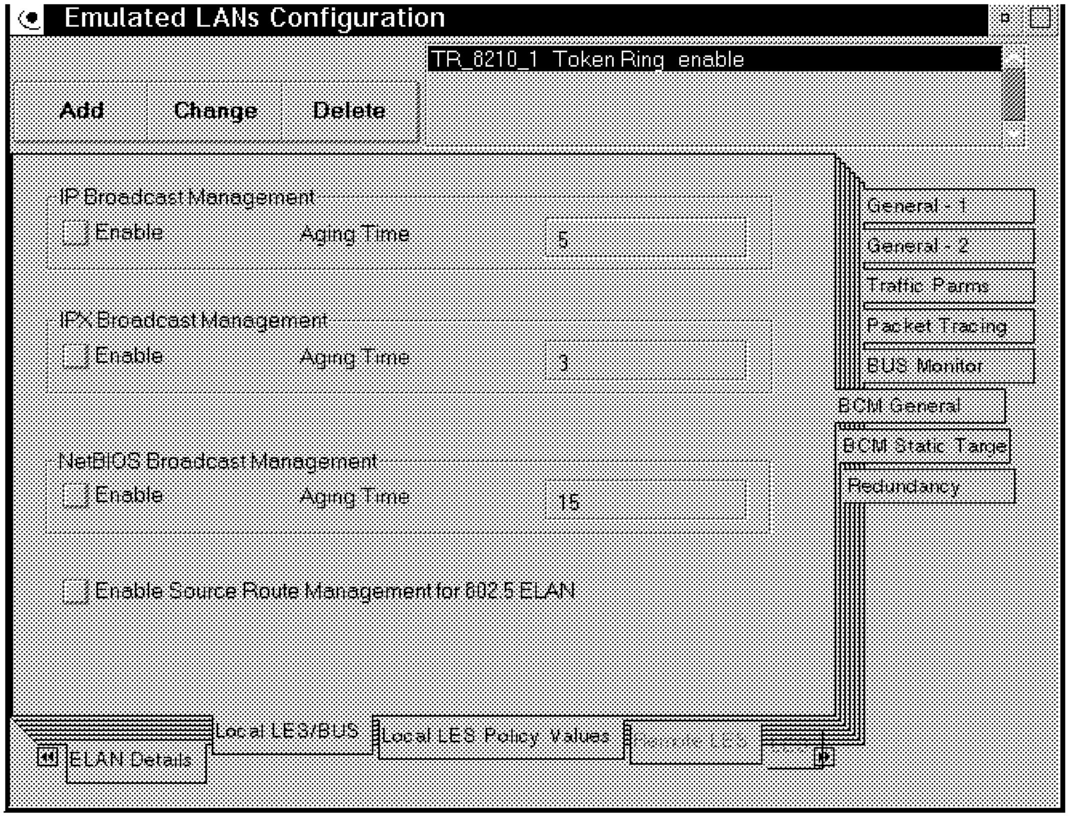
Frame Sampling Rate

10

Figure 58. Emulated LAN Configuration - Local LES/BUS. Activate BUS monitoring.

5 Broadcast manager (BCM) and source route manager (SRM)

Select BCM General to configure BCM and SRM functions as depicted in Figure 59.



TR_8210_1 Token Ring enable

IP Broadcast Management
□ Enable Aging Time 5

IPX Broadcast Management
□ Enable Aging Time 3

NetBIOS Broadcast Management
□ Enable Aging Time 15

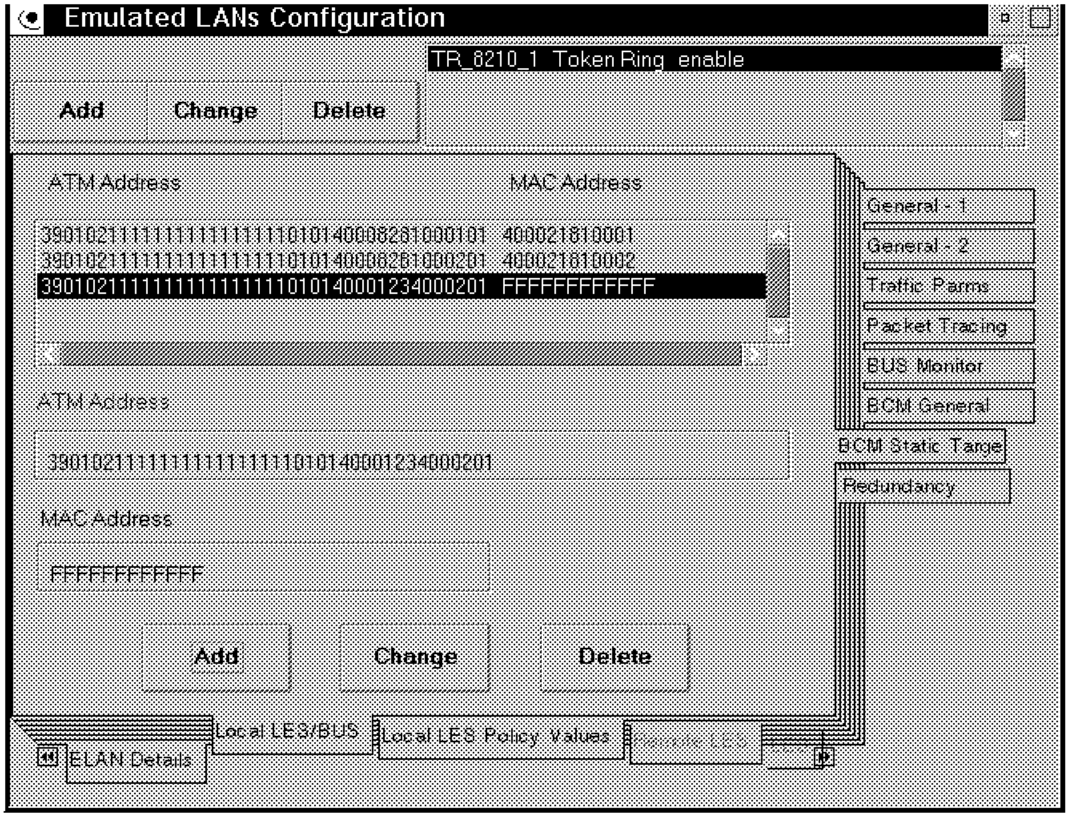
□ Enable Source Route Management for 802.5 ELAN

Figure 59. Emulated LAN Configuration - Local LES/BUS. Activate BCM and SRM.

BCM functions can be selected for IP, IPX, and/or NetBIOS. Note that SRM, enabled via Enable Source Route Management for 802.5 ELAN, only operates on protocols that have been enabled for BCM.

6 BCM static definitions

BCM static definitions, see Figure 60, are required for IPX stations that do not broadcast RIP and/or SAP messages, but need to receive RIP/SAP messages from other IPX servers or servers farm (see 5.2.4.2, “Static IPX/BCM Definitions - Servers Farms” on page 98).



TR_8210_1 Token Ring enable

ATM Address	MAC Address
3901021111111111111111010140008281000101	400021810001
3901021111111111111111010140008281000201	400021810002
3901021111111111111111010140001234000201	FFFFFFFFFFFF

ATM Address: 3901021111111111111111010140001234000201

MAC Address: FFFFFFFFFFFF

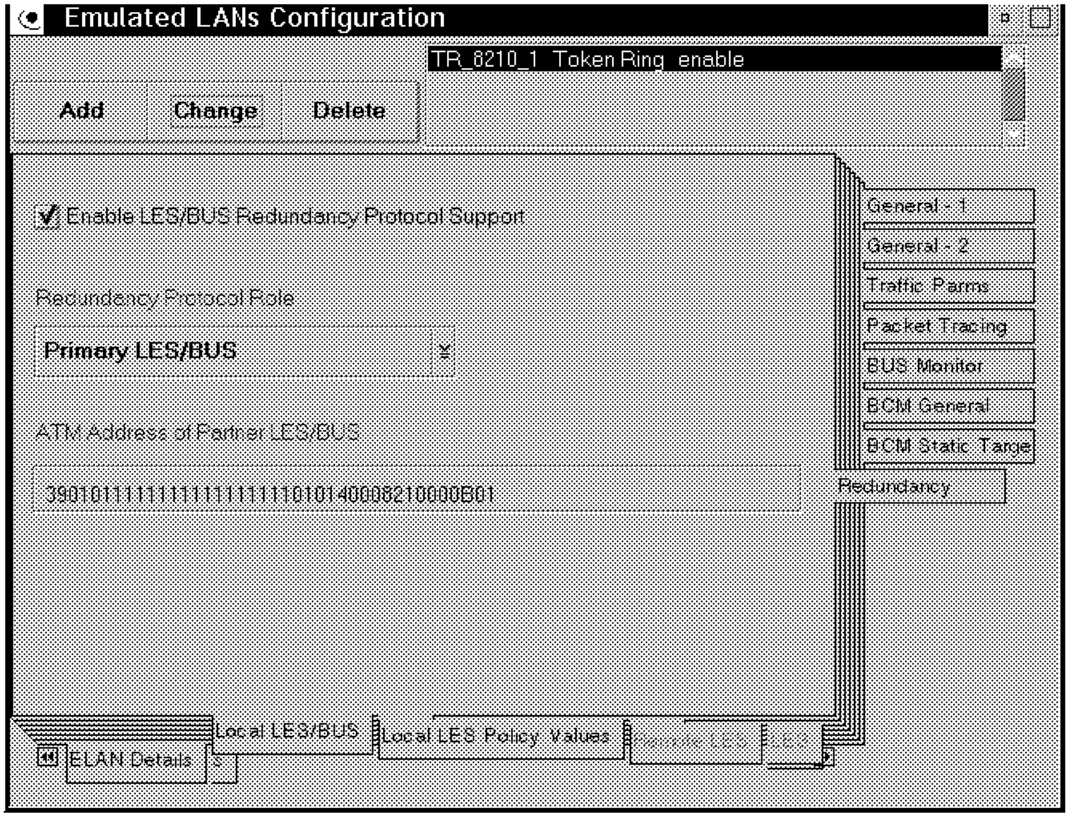
Figure 60. Emulated LAN Configuration - Local LES/BUS. BCM static targets.

Each static entry contains an LE client ATM address and a valid MAC address (either unicast or X′FFFFFFFFFFFF′). Up to three static entries can be defined.

Note: This step requires that the MSS Server has enabled the BCM functions for IPX traffic.

7 LES/BUS redundancy

Clicking on the Redundancy button during Local LES/BUS configuration enables you to configure the LECS/LES security interface (see Figure 61).



TR_8210_1 Token Ring enable

☑ Enable LES/BUS Redundancy Protocol Support

Redundancy Protocol Role: Primary LES/BUS

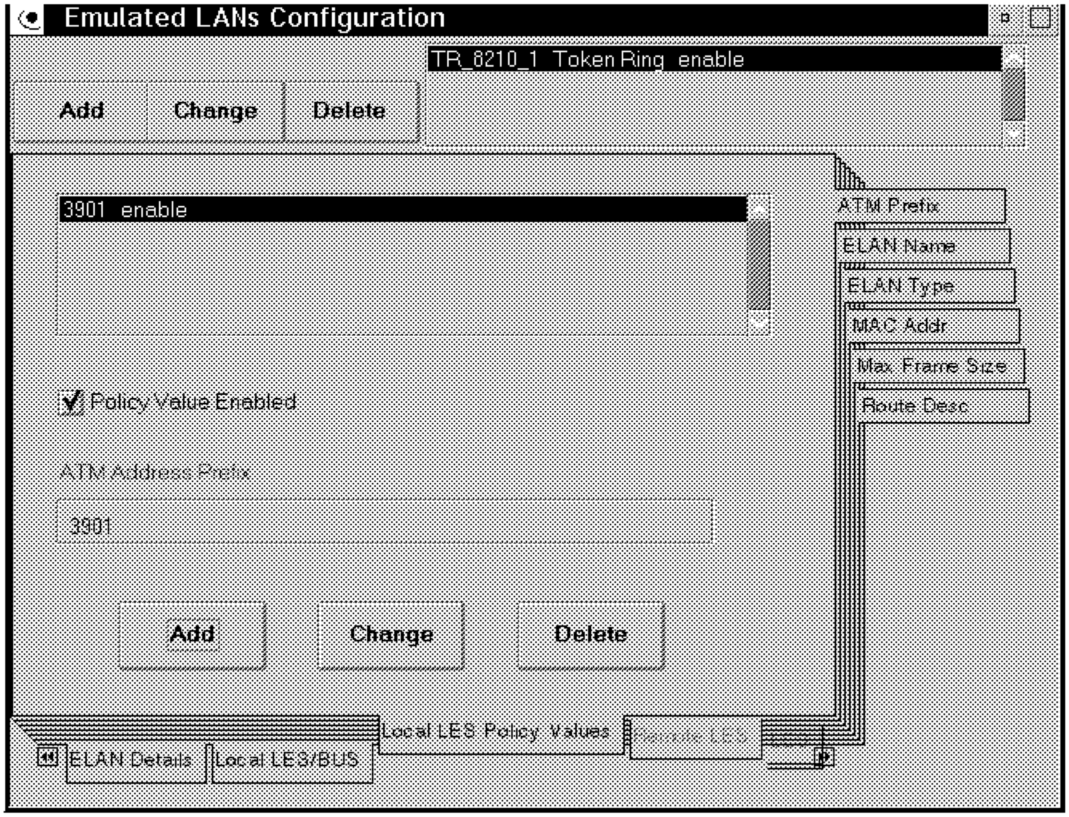
ATM Address of Partner LES/BUS: 3901011111111111111111010140008210000B01

Figure 61. Emulated LAN Configuration - Local LES/BUS. LES/BUS redundancy.

Specify if LES/BUS redundancy has to be enabled. If yes, indicate whether the local LES/BUS is primary or backup. If primary, indicate the ATM address of the remote backup LES/BUS.

8 ELAN policy values

LE clients are only assigned to an ELAN if a best match occurs between an enabled LECS policy *type* and an ELAN defined policy *value*. Figure 62 becomes available after selecting Local LES Policy Values. In this screen you can define the policy values relevant for this ELAN. The screen depicts an ATM prefix value, but other values can be added.



Emulated LANs Configuration

TR_8210_1 Token Ring enable

3901 enable

☑ Policy Value Enabled

ATM Address Prefix

3901

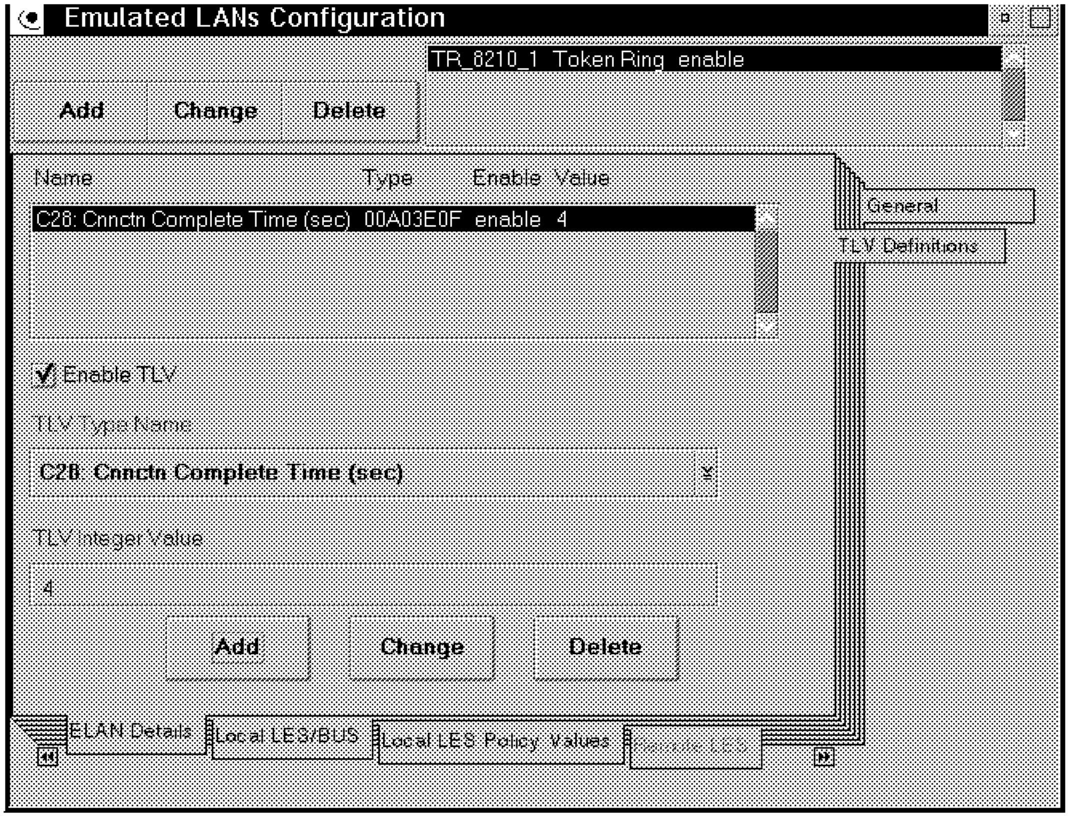
Figure 62. Emulated LAN Configuration - Local LES Policy Value. ATM address prefix.

Policy values can be added, modified, or deleted using the Add, Change, and Delete buttons at the bottom. Click on Change at the top to associate new or modified values with the ELAN being configured.

Policy values for this ELAN are only required when the LECS function has been activated on the MSS Server. Make sure that the policy values are not ambiguous (no conflicts between ELANs), and that you define values for policy types that have been enabled during your LECS definition (see 2 on page 136).

9 Define TLVs for this ELAN

Selecting ELAN Details and clicking on TLV Definitions results in Figure 63 appearing.



TR_8210_1 Token Ring enable

Name	Type	Enable	Value
C28: Cnnctn Complete Time (sec)	00A03E0F	enable	4

☑ Enable TLV

TLV Type Name: C28: Cnnctn Complete Time (sec)

TLV Integer Value: 4

Figure 63. Emulated LAN Configuration - Local LES Policy Value. TLV Definitions.

This screen enables you to set the TLV values the LECS will return when LE clients connect to it.

Notes:

a. When you want to modify the system-defined TLVs or add user-defined TLVs, see 5.11.4, "Defining and Modifying TLVs" on page 140.

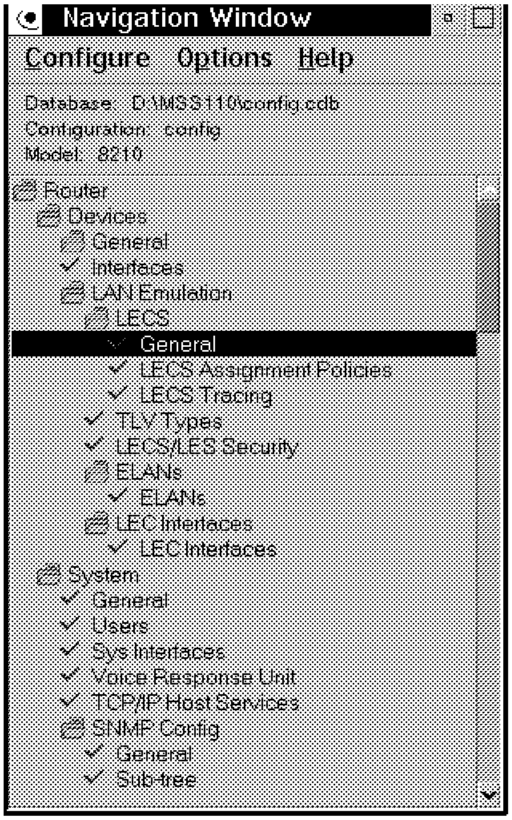
b. Specify TLVs only when a local LECS is active for this ELAN.

5.11.2 Configuring LECS

Definition steps during LECS configuration are:

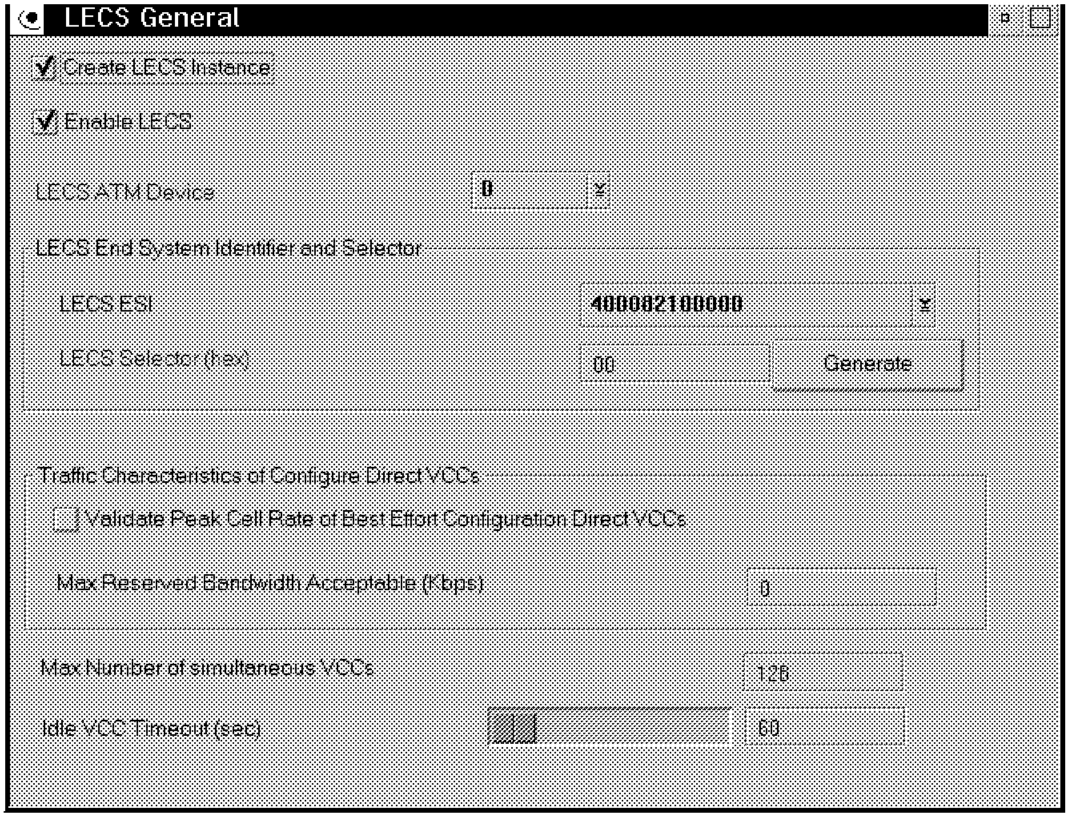
1 Create and enable LECS instance

Configuring the LECS function can be done by clicking on General in the Navigation Window, see Figure 64 on page 134, after which Figure 65 on page 135 appears.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 64. LECS Definition



LECS General

☑ Create LECS Instance

☑ Enable LECS

LECS ATM Device: 0

LECS End System Identifier and Selector

LECS ESI: 400082100000

LECS Selector (hex): 00

Traffic Characteristics of Configure Direct VCCs

☐ Validate Peak Cell Rate of Best Effort Configuration Direct VCCs

Max Reserved Bandwidth Acceptable (Kbps): 0

Max Number of simultaneous VCCs: 128

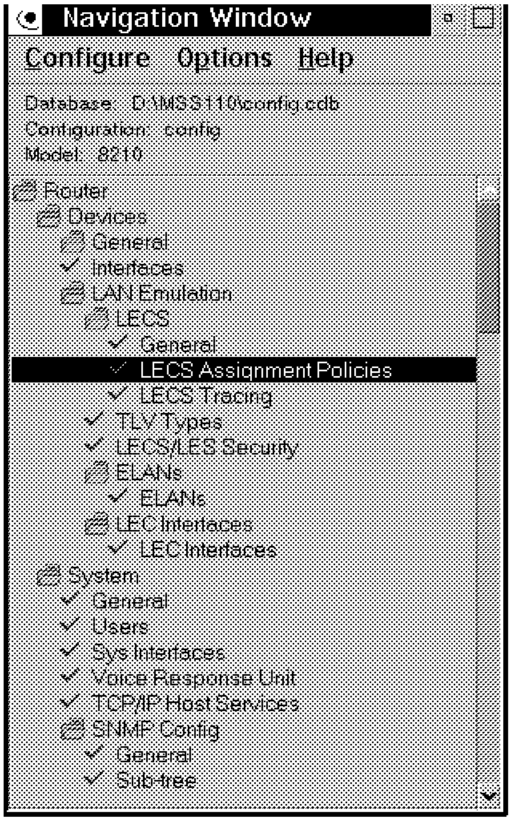
Idle VCC Timeout (sec): 60

Figure 65. LECS General

Clicking on Create LECS Instance and Enable LECS results in an active LECS. Associate the LECS with one of the ATM ports, and select the ESI and SEL used for constructing the ATM address used by this LECS. It is advised that you use a locally administered ESI. To prevent the SEL byte from changing in future configurations, it is advised that you use SEL=0, as generated by the configurator.

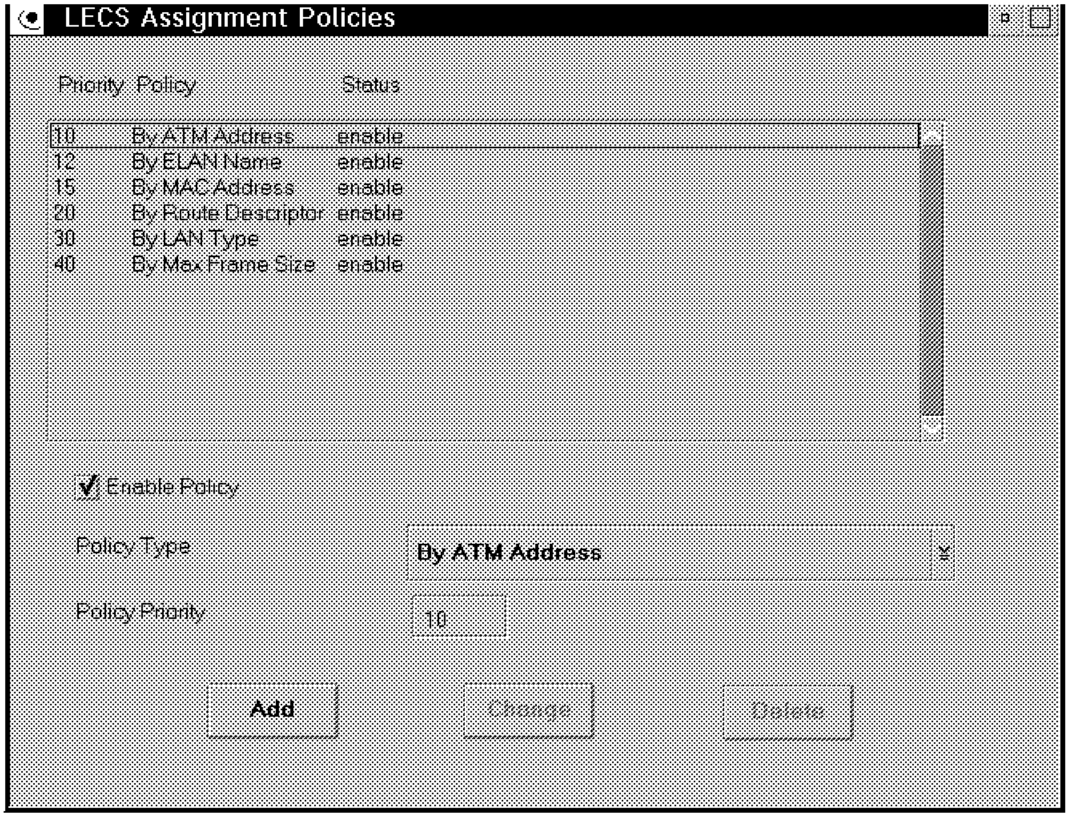
2 Define LECS policies/priority

Selecting LECS Assignment Policies in the Navigation Window (see Figure 66) enables you to define the LECS assignment policy types and the priority order in which the policies are applied.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 66. Defining LECS Assignment Policies



LECS Assignment Policies

Priority	Policy	Status
10	By ATM Address	enable
12	By ELAN Name	enable
15	By MAC Address	enable
20	By Route Descriptor	enable
30	By LAN Type	enable
40	By Max Frame Size	enable

☑ Enable Policy

Policy Type: By ATM Address

Policy Priority: 10

Figure 67. LECS Assignment Policies

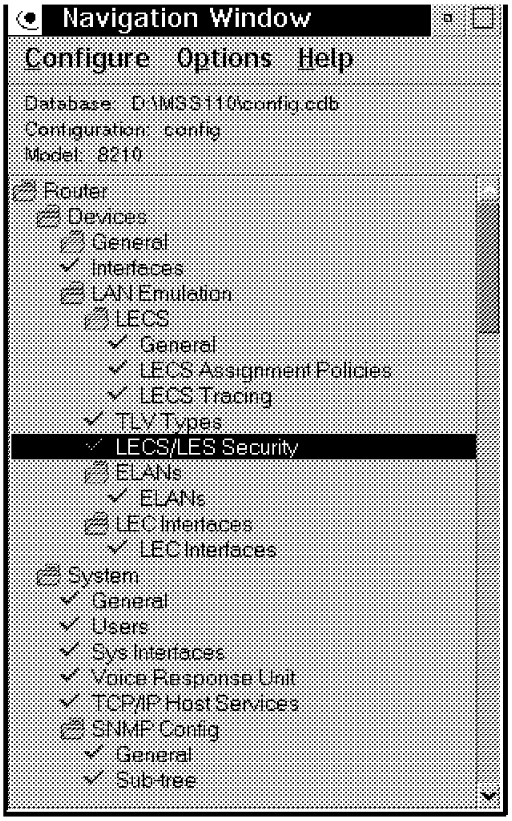
Figure 67 enables you to define the policy types and their priority, which are used by the LECS to assign LE clients to ELANs. Multiple, up to six, policy types can be enabled. Note that a high number indicates a low priority.

Note: The definition of LECS policy types needs to be done in conjunction with the policy values defined during ELAN definition (see 8 on page 132).

As your MSS Server can have a single LECS instance only, the LECS configuration steps have to be done only once.

5.11.3 Configuring LECS/LES Security Interface

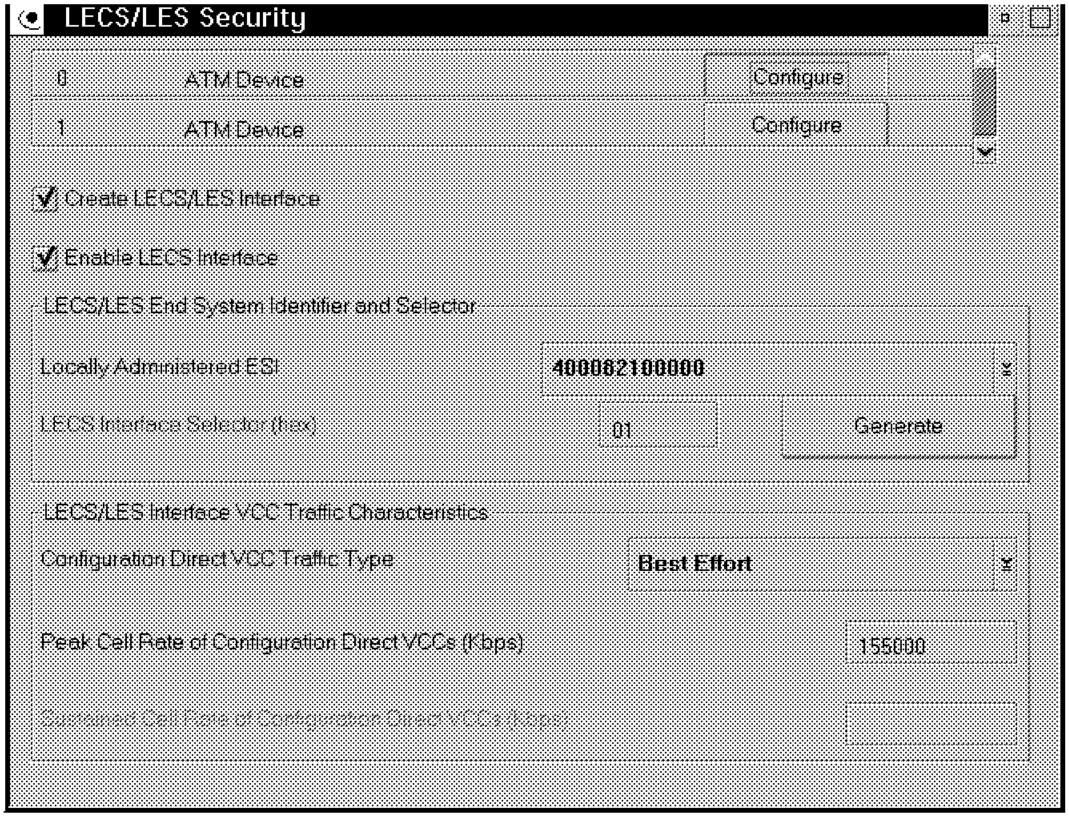
The LECS/LES security feature needs to be enabled if any of your ELANs have been defined as a secure ELAN. Select LECS/LES Security from the Navigation Window to obtain Figure 69 on page 139.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 68. Defining LECS/LES Security

Note: Defining the security interface is mandatory if you have defined at least one secure ELAN. This step is only required once per MSS Server.



LECS/LES Security

0 ATM Device

1 ATM Device

☑ Create LECS/LES Interface

☑ Enable LECS Interface

LECS/LES End System Identifier and Selector

Locally Administered ESI 400082100000

LECS Interface Selector (hex) 01

LECS/LES Interface VCC Traffic Characteristics

Configuration Direct VCC Traffic Type Best Effort

Peak Cell Rate of Configuration Direct VCCs (Kbps) 155000

Sustained Cell Rate of Configuration Direct VCCs (Kbps)

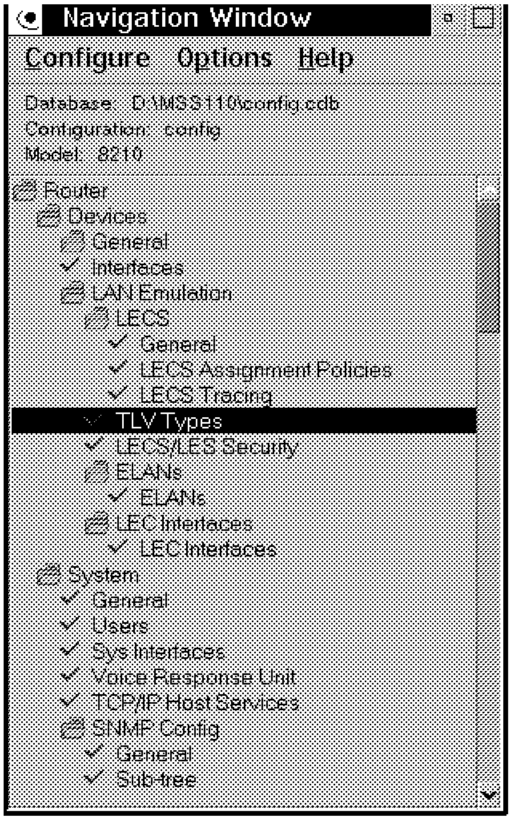
Figure 69. LECS/LES Security

Create and enable the security interface to activate it. During its definition associate the security interface with an ATM port, assign an ESI and SEL to it, and define the type of configuration direct VCC that should be used to the remote LECS.

5.11.4 Defining and Modifying TLVs

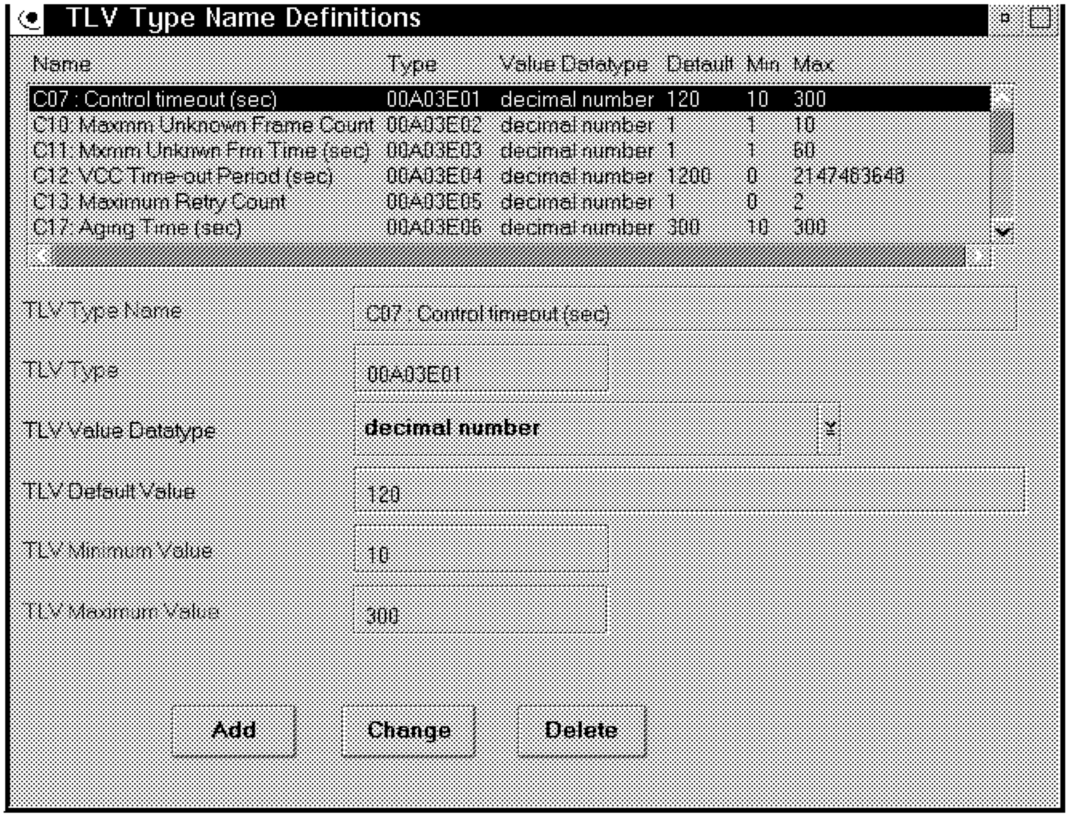
During the definition of your ELANs you can specify which TLVs are returned by the LECS to its LE clients. Hereby you can select (see 9 on page 133) user-defined or system-defined TLVs.

If you want to modify the system-defined TLVs or add user-defined TLVs, click on TLV Types within the Navigation Window (see Figure 70). Figure 71 on page 141 results.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 70. TLV Types



Name	Type	Value Datatype	Default	Min	Max
C07 : Control timeout (sec)	00A03E01	decimal number	120	10	300
C10: Maxmm Unknown Frame Count	00A03E02	decimal number	1	1	10
C11: Mxmm Unknwn Frm Time (sec)	00A03E03	decimal number	1	1	60
C12: VCC Time-out Period (sec)	00A03E04	decimal number	1200	0	2147483648
C13: Maximum Retry Count	00A03E05	decimal number	1	0	2
C17: Aging Time (sec)	00A03E06	decimal number	300	10	300

TLV Type Name: C07 : Control timeout (sec)

TLV Type: 00A03E01

TLV Value Datatype: decimal number

TLV Default Value: 120

TLV Minimum Value: 10

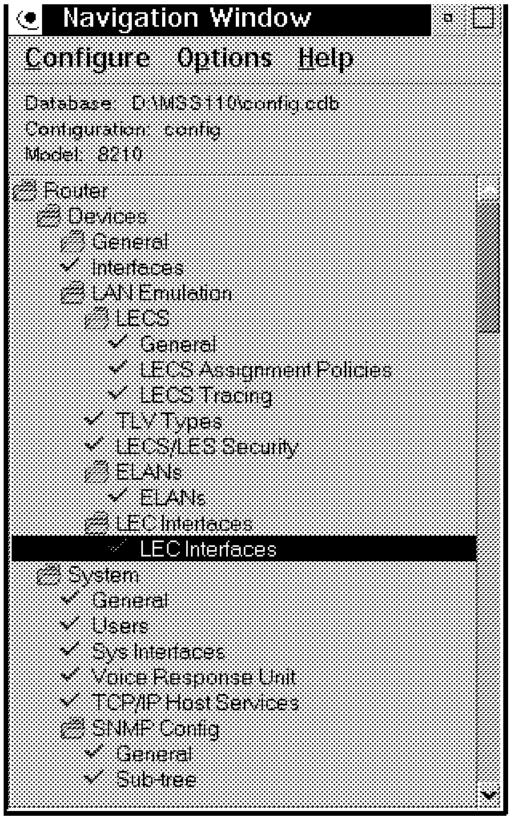
TLV Maximum Value: 300

Figure 71. TLV Type Name Definitions

Figure 71 enables you to add, modify or delete TLVs returned by the LECS when LE clients connect to it.

5.11.5 Configuring LE Clients

The configuration of an LE client starts by selecting LEC Interfaces in the Navigation Window, as shown in Figure 72.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

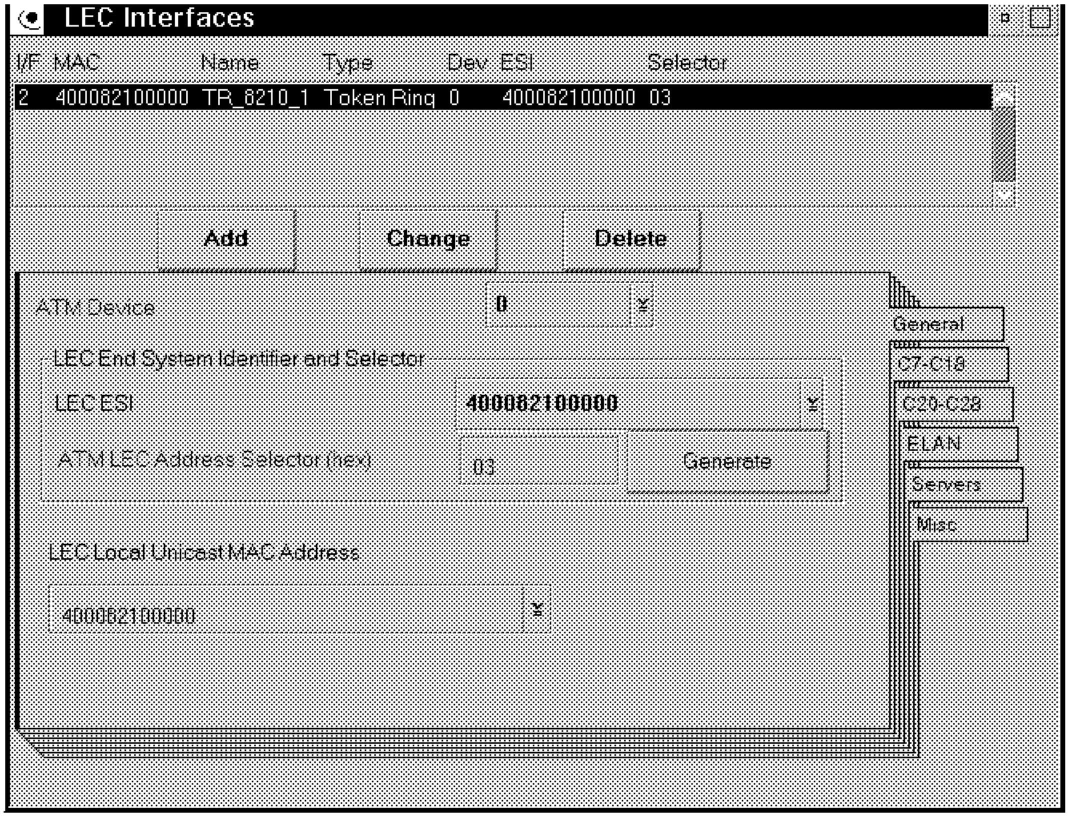
Figure 72. Define LEC Interfaces

Note: The configuration steps discussed in this section need to be repeated for every LE client that has to be defined on your MSS Server.

The configuration steps required are:

1 Define LE client addresses

After selecting LEC Interfaces from the Navigation Window, Figure 73 appears.



LEC Interfaces

I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100000	TR_8210_1	Token Ring	0	400082100000	03

ATM Device: 0

LEC End System Identifier and Selector

LEC ESI: 400082100000

ATM LEC Address Selector (hex): 03

LEC Local Unicast MAC Address: 400082100000

Figure 73. LEC Interfaces

During LE client definition, indicate the ATM port it is associated with, the ESI and SEL used to construct the LE client′s ATM address, and the MAC address associated with the LE client. To simplify problem determination, a locally administered ESI is recommended. Use an SEL that is generated by the configurator. Make sure the MAC address is unique.

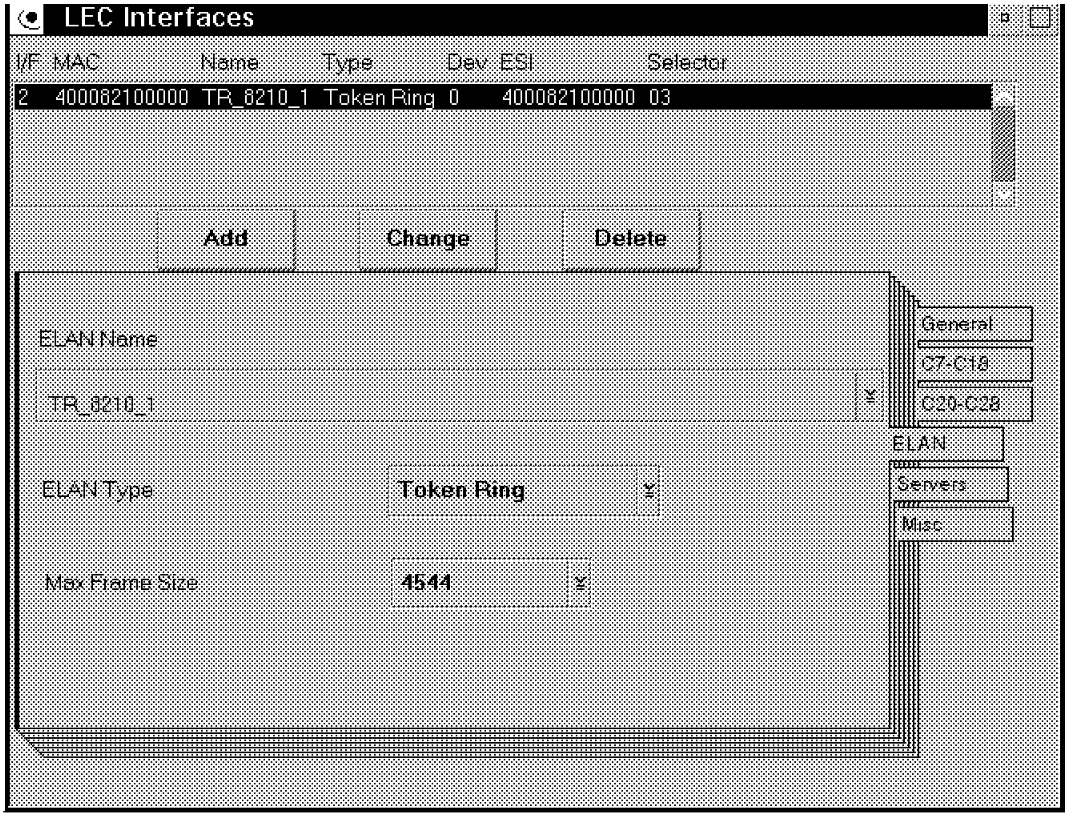
Notes:

a. LE clients can be ATM Forum or IBM compliant. The MSS Server Release 1.0 code provides support for ATM Forum-compliant LE services only.

b. When the LE client is added, a logical interface number will be generated. This interface number (I/F) is required when configuring higher-layer functions such as IP or bridging for this LEC.

2 Define the ELAN name and type

After selecting ELAN in Figure 73 on page 143, Figure 74 appears.



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100000	TR_8210_1	Token Ring	0	400082100000	03

ELAN Name: TR_8210_1

ELAN Type: Token Ring

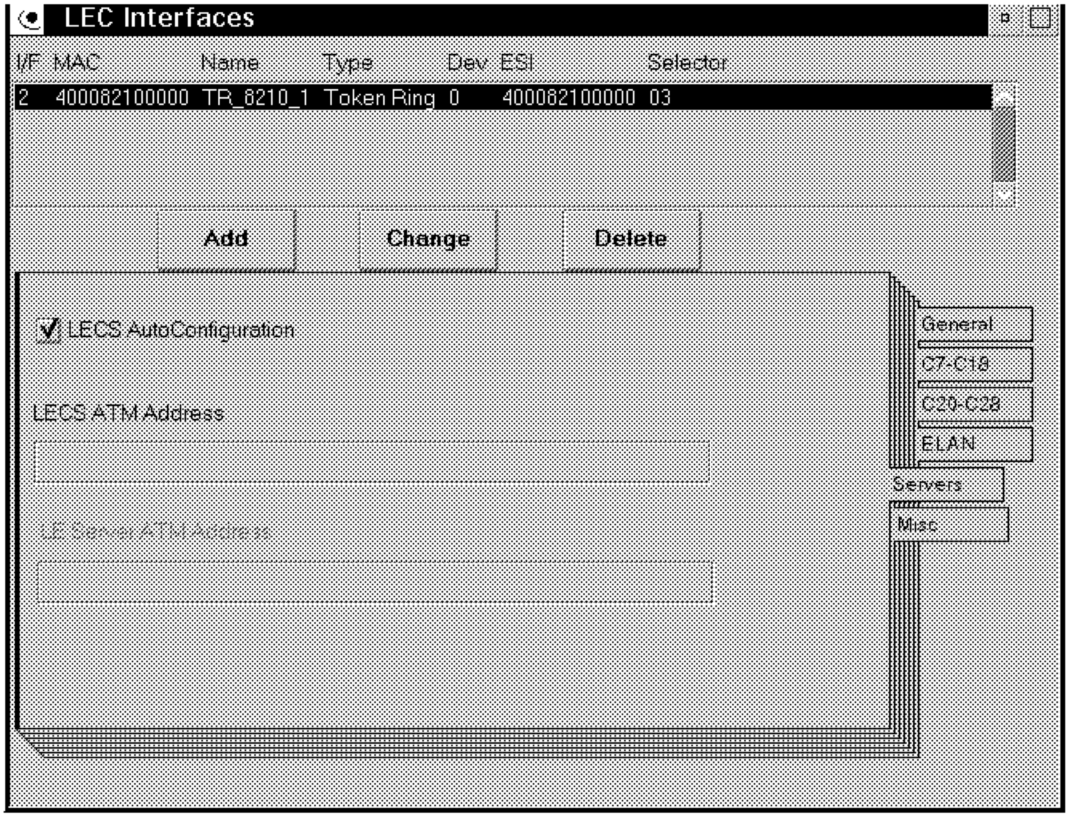
Max Frame Size: 4544

Figure 74. LEC Interfaces - ELAN

It is mandatory that you specify the ELAN type (token-ring or Ethernet) and maximum frame size. The ELAN name is optional. It is advised that you define the same name on the LE client as defined on LECS and LES.

3 Define LE servers

LE clients either obtain their LES address from the LECS or use a hard-coded LES ATM address. The LECS can be hard-coded or, using ILMI, learned from the adjacent ATM switch.



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100000	TR_8210_1	Token Ring	0	400082100000	03

☑ LECS AutoConfiguration

LECS ATM Address

LE Server ATM Address

Figure 75. LEC Interfaces - LECS Auto-Configuration

Figure 75 results from selecting Server during LEC interface definition. LECS Autoconfiguration specifies that the LE client will learn the LECS address from the adjacent ATM switch. Alternatively, a hard-coded LECS or hard-coded LES ATM address can be specified.

4 Define higher-layer functions

In addition to the basic LE client configuration steps listed earlier, configuration for higher-layer (bridging or routing) functions is required before the LE client can be used. For details on IP routing, IPX routing, and bridging, see Chapter 7, “MSS Server and IP Routing Protocols” on page 169, Chapter 8, “MSS Server and IPX Routing” on page 205, and Chapter 9, “MSS Server and Bridging” on page 233.

Chapter 6. MSS Server and Classical IP

Classical IP, as described in RFC 1577, defines the operation of IP and the address resolution protocol (ARP) over ATM. It also defines the initial application of ATM within Classical IP networks as a direct replacement for local area networks (Ethernet and token-ring) and for IP links that interconnect routers, either within or between administrative domains. The classical model here refers to the treatment of the ATM host adapter as a networking interface to the IP protocol stack operating in a LAN-based environment.

IBM has been delivering this functionality for the RS/6000 since March of 1994 and is continuing this with the introduction of the IBM 8210 Nways MSS Server. In the following sections we give an short introduction to Classical IP. IBM 8210 Nways MSS Server implementation and configuration details are discussed in 6.2, “Using Classical IP on the MSS Server” on page 153.

6.1 Introduction to Classical IP

Some characteristics of the Classical IP model, which are discussed in more detail below, are the following:

- The Logical IP Subnetwork (LIS)
- ATM Address Resolution Protocol - ATMARP
- ATMARP servers

  Note: In this publication we use the terms ATMARP server and ARP server interchangeably.
- ATMARP clients

  Note: In this publication we use the terms ATMARP client and LIS client interchangeably.
- ATMARP tables
- ATMARP packet formats
- ATMARP/InATMARP packet encapsulation

RFC 1577 does not describe the operation of ATM networks. Any reference to virtual connections, permanent virtual connections, or switched virtual connections applies only to virtual channel connections used to support IP and address resolution over ATM, and thus are assumed to be using AAL5.

6.1.1 Logical IP Subnetwork Configuration

In the LIS scenario, each separate administrator configures hosts and routers within a closed logical IP subnetwork. Each LIS operates and communicates independently of other LISs on the same ATM network. Hosts connected to ATM communicate directly to other hosts within the same LIS. Communication to hosts outside of the local LIS is provided via an IP router. This router is an ATM end system attached to the ATM network that is configured as a member of two or more LISs. This configuration may result in a number of disjoint LISs operating over the same ATM network. Hosts of differing IP subnets communicate via an intermediate IP router even though it may be possible to open a direct VCC between the two IP members over the ATM network. This is illustrated in Figure 76 on page 148.

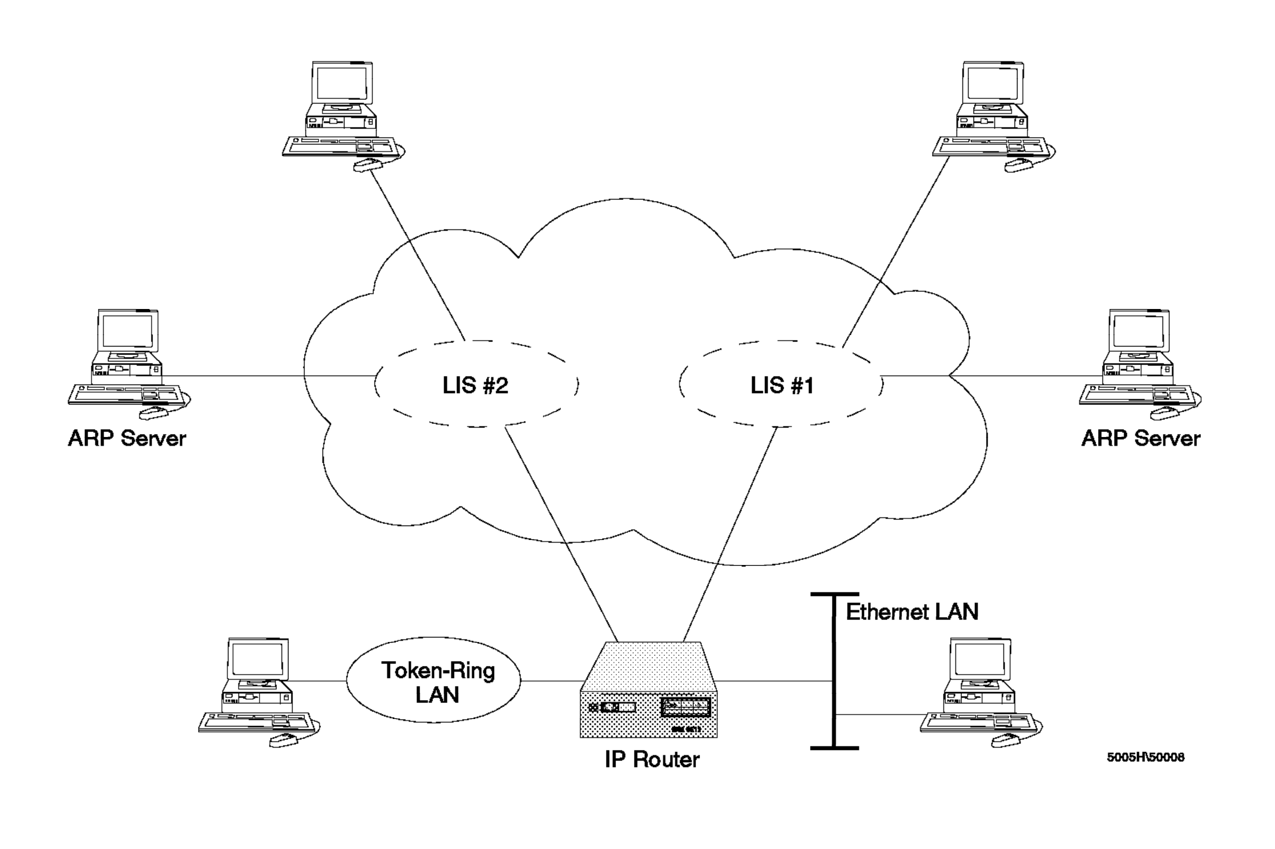


Figure 76. Logical IP Subnetworks

The following are the requirements for all IP members (hosts, routers) operating in an ATM LIS configuration:

- The same IP network/subnet number and address mask.
- A direct connection to the ATM network.
- The same maximum tranfer unit (MTU) for all members within a LIS.

  Note: The MTU is the maximum AAL-5 service data unit (SDU) minus 8. See also item 2e on page 155.
- All members outside of the LIS are accessed via a router.
- A service for resolving IP addresses to ATM addresses via ATMARP and vice versa via InATMARP when using SVCs.
- A service for resolving VCCs to IP addresses via InATMARP when using PVCs.
- To be able to communicate via ATM with all other members in the same LIS, the virtual connection topology underlying the intercommunication among the members is fully meshed.

6.1.2 Address Resolution

Address resolution within an ATM logical IP subnet makes use of the ATM Address Resolution Protocol (ATMARP) and the Inverse ATM Address Resolution Protocol (InATMARP). ATMARP is the same protocol as the IP ARP protocol with extensions needed to support ARP in a unicast server ATM environment. InATMARP is the same protocol as the original IP InARP protocol but applied to ATM networks. Protocol fields have been updated to contain the ATM address information. All IP stations must support these protocols as they are revised in future versions of the RFC. Use of these protocols is the same, however, fields may be different depending on whether PVCs or SVCs are used.

6.1.2.1 Switched Virtual Connections with ARP Server

The most common way to support IP over ATM Switched Virtual Connections (SVCs) is to use an ATMARP server within your LIS. For each LIS a different ATMARP server instance is required. Note that the IBM 8210 Nways MSS Server supports multiple ATMARP server instances per ATM port.

This server will be responsible for resolving the ATMARP requests of all IP members within the LIS.

To use the server, each member of the LIS must register at the ATMARP server first. To do this, an individual client connects to the ATMARP server using a point-to-point VCC. Once the connection between the client and the ATMARP server is established, the server transmits an InARP_REQUEST to determine the IP address of the client. The InARP_REPLY from the client contains the information necessary for the ATMARP server to build its ATMARP table cache. This information is used by the server to generate replies to the ATMARP requests it receives from the clients.

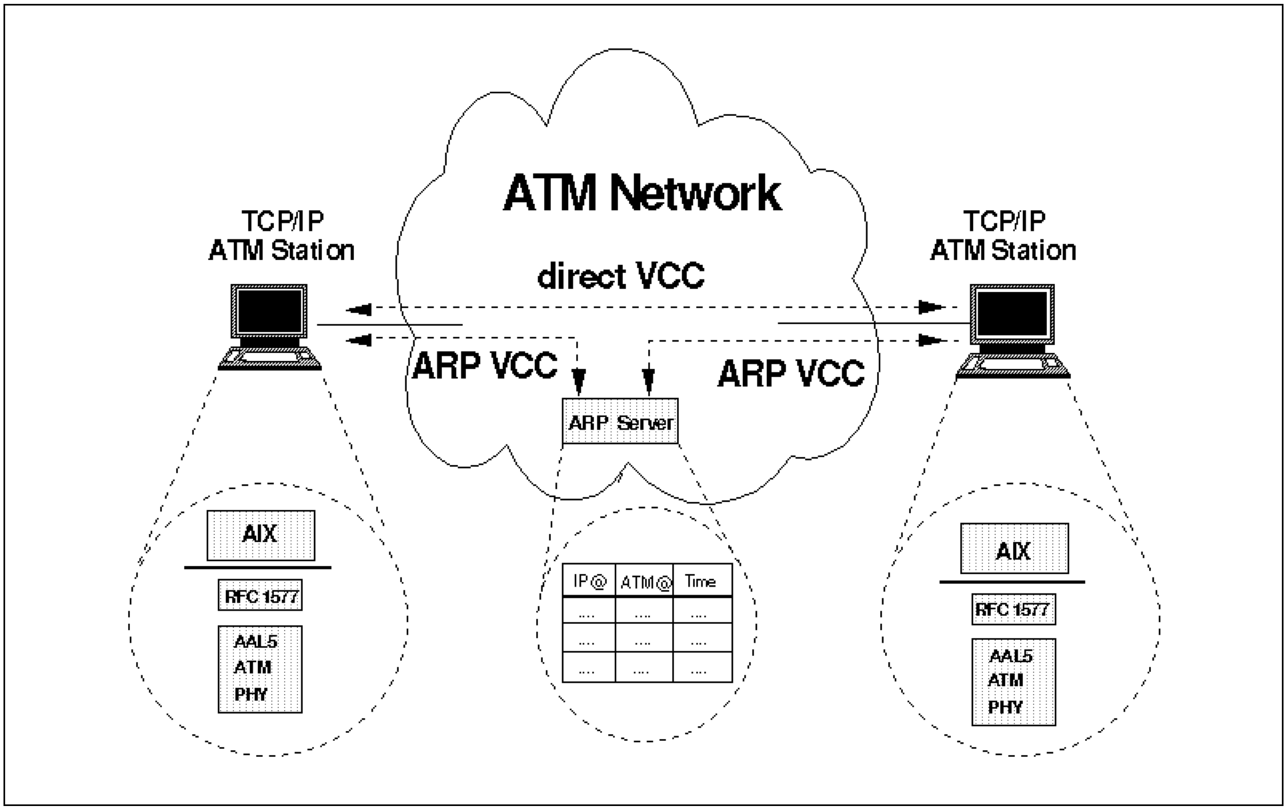


Figure 77. Classic IP over ATM

After registration with the ATMARP server, the client may send ATMARP requests to the ATMARP server in order to find out the ATM address of the other clients when it needs to communicate with them. Once the ATM address of the destination is known, the client must establish a direct point-to-point VCC with the destination in order to communicate with it. Note that the ATMARP server should only be used as a directory server and not to forward the frames exchanged between two clients.

Note: An ARP server on the 8210 will forward IP frames between two clients of the same LIS, however, will sent an ICMP redirect message.

The ATMARP server mechanism requires that each client be administratively configured with the ATM address of the ATMARP server.

Switched Virtual Connections without ARP Server: The IBM 8210 Nways MSS Server introduces an alternative way of using SVCs to establish data connections between IP members within a LIS, which does not necessitate an ARP server. Instead of using an ARP server to learn the ATM address of a partner client on the same LIS, an option exists to predefine ARP entries for remote LIS clients.

Advantage of this method is that no ARP server is required, and because the LIS clients use SVCs, call establishment benefits from the dynamics in the ATM network. The disadvantage is that partner LIS clients have to be predefined.

6.1.2.2 Permanent Virtual Connections

When PVCs are used to connect workstations together, there is no need for an ARP server. However, each IP station must have a mechanism (for example, manual configuration) for determining what PVCs it has and, in particular, which PVCs are being used with LLC/SNAP encapsulation.

Each member of the LIS is required to use the Inverse ATM Address Resolution Protocol (InATMARP) on those VCs to determine the IP address of the station at the other end of the PVC (InARP_REQUEST). When the requesting station receives the InARP_REPLY, it may complete the ATMARP table entry and use the provided address information.

Note: Information learned via InARP_REPLY may be aged or invalidated under certain circumstances. It is the responsibility of each IP station supporting PVCs to revalidate ATMARP table entries as part of the aging process. The MSS Server will not validate pre-configured partner IP addresses on PVCs.

6.1.3 ATMARP Server Operational Requirements

The ATMARP server accepts ATM calls/connections from other ATM end systems. At call setup, if the VCC supports LLC/SNAP encapsulation, the ATMARP server will transmit to the originating ATM station an InARP_REQUEST for each logical IP subnet the server is configured to serve. After receiving an InARP_REPLY, the server will examine the IP address and the ATM address. The server will add (or update) the ATM address and IP address map entry and time stamp into its ATMARP table. If the IP address received in the InARP_REPLY duplicates a table entry′s IP address, the ATM address does not match the table entry′s ATM address, and there is an open VCC associated with that table entry, the InARP_REPLY information is discarded and no modifications to the table are made. ATMARP table entries persist until aged or invalidated. VCC call tear down does not remove ATMARP table entries.

The ATMARP server, upon receiving an ARP_REQUEST, will generate the corresponding ARP_REPLY if it has an entry in its ATMARP table; otherwise, it will generate a negative reply (ARP_NAK). The ARP_NAK response is an extension to the ATMARP protocol and is used to improve the robustness of the ATMARP server mechanism. With ARP_NAK, a client can determine the difference between a catastrophic server failure and an ATMARP table lookup failure. The ARP_NAK packet format is the same as the received ARP_REQUEST packet format with the operation code set to ARP_NAK (that is, the ARP_REQUEST packet data is merely copied for transmission with the ARP_REQUEST operation code reset to ARP_NAK).

When the server receives an ATMARP request over a VCC where the source IP and ATM address match the association already in the ATMARP table, and the ATM address matches that associated with the VC, the server may update the timeout on the source ATMARP table entry. For example, if the client is sending ATMARP requests to the server over the same VCC that it used to register its ATMARP entry, the server should examine the ATMARP requests and note that the client is still alive by updating the timeout on the client′s ATMARP table entry.

It is obvious that the ATMARP server is a critical component in creating and maintaining LIS connectivity. See 6.3.5, "Redundant ARP Server" on page 166 to learn how two IBM 8210 Nways MSS Servers can be used to provide redundancy.

6.1.4 ATMARP Client Operational Requirements

The ATMARP client is responsible for contacting the ATMARP server to register its own ATMARP information and to gain and refresh its own ATMARP entry/information about other IP members. This means, as noted above, that ATMARP clients must be configured with the ATM address of the ATMARP server. ATMARP clients need to do the following:

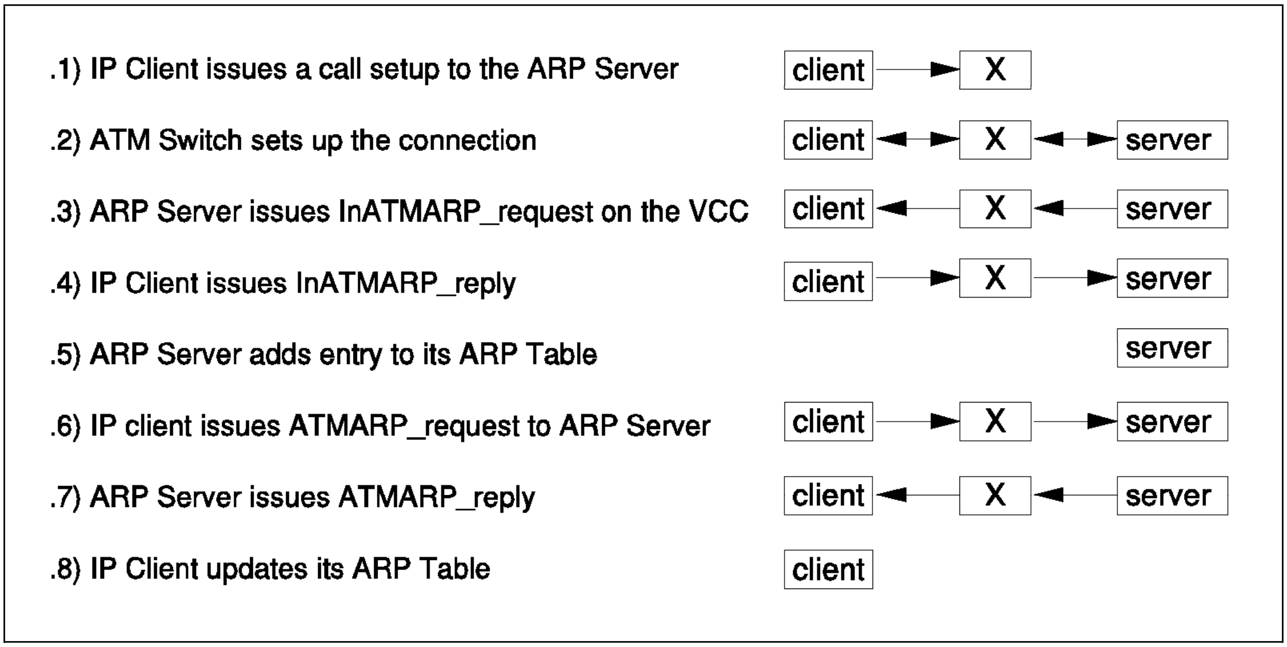
1. Initiate the VCC connection to the ATMARP server for transmitting and receiving ARP_REQUEST and InARP_REQUEST packets.

2. Respond to ARP_REQUEST and InARP_REQUEST packets received on any VCC appropriately.

3. Generate and transmit ARP_REQUEST packets to the ATMARP server and process ARP_REPLY and ARP_NAK packets from the server appropriately. ARP_REPLY packets should be used to build/refresh the client′s own ATMARP table entries.

4. Generate and transmit InARP_REQUEST packets as needed and process InARP_REPLY packets appropriately. InARP_REPLY packets will be used by the client to build/refresh its own client ATMARP table entries.

5. Provide an ATMARP table aging function to remove its own old ATMARP tables entries after a convenient period of time.



.1) IP Client issues a call setup to the ARP Server

.2) ATM Switch sets up the connection

.3) ARP Server issues InATMARP_request on the VCC

.4) IP Client issues InATMARP_reply

.5) ARP Server adds entry to its ARP Table

.6) IP client issues ATMARP_request to ARP Server

.7) ARP Server issues ATMARP_reply

.8) IP Client updates its ARP Table

Figure 78. IP Address Resolution Scenario

Note: If the client does not maintain an open VCC to the server, the client must refresh its ATMARP information with the server at least once every 20 minutes. This is done by opening a VCC to the server and exchanging the initial InATMARP packets.

6.1.5 ATMARP Table Aging

An ATMARP client or server must know about any open VCCs it has (permanent or switched), their association with an ATMARP table entry, and which VCs support LLC/SNAP encapsulation.

Client ATMARP table entries are valid for a maximum time of 15 minutes. Default value on the MSS Server is 5 minutes

Server ATMARP table entries are valid for a minimum time of 20 minutes. Default value on the MSS Server is 20 minutes

Prior to aging an ATMARP table entry, an ATMARP server generates an InARP_REQUEST on any open VCC associated with that entry. If an InARP_REPLY is received, that table entry is updated and not deleted. If there is no open VCC associated with the table entry, the entry is deleted.

When an ATMARP table entry ages, an ATMARP client invalidates the table entry. If there is no open VCC associated with the invalidated entry, that entry is deleted. In the case of an invalidated entry and an open VCC, the ATMARP client revalidates the entry prior to transmitting any non-address resolution traffic on that VCC. In the case of a PVC, the client validates the entry by transmitting an InARP_REQUEST and updating the entry on receipt of an InARP_REPLY. In the case of an SVC, the client validates the entry by transmitting an ARP_REQUEST to the ATMARP server and updating the entry on receipt of an ARP_REPLY. If a VCC with an associated invalidated ATMARP table entry is closed, that table entry is removed.

Note: The MSS Server provides a configuration option ("Enable auto-refresh") to enable auto-refresh of table entries. This option is by default disabled for LIS clients. A second configuration option ("Refresh by InATMARP") exist to validate table entries at remote clients instead of the ARP server. This decreases the dependency on the ARP server.

6.1.6 ATMARP and InATMARP Packet Format

Internet addresses are assigned independently of ATM addresses. Each host implementation knows its own IP and ATM address(es) and responds to address resolution requests appropriately. IP members also use ATMARP and InATMARP to resolve IP addresses to ATM addresses when needed.

The ATMARP and InATMARP protocols use the same protocol type, and operation code data formats as the ARP and InARP protocols. The location of these fields within the ATMARP packet are in the same byte position as those in ARP and InARP packets. A unique hardware type value (X′13′) has been assigned for ATMARP. In addition, ATMARP makes use of an additional operation code for ARP_NAK. The remainder of the ATMARP/InATMARP packet format is different than the ARP/InARP packet format.

6.1.7 ATMARP/InATMARP Packet Encapsulation

ATMARP and InATMARP packets are to be encoded in AAL5 PDUs using the LLC/SNAP encapsulation described in RFC 1483.

The LLC value of 0xAA-AA-03 (3 octets) indicates the presence of a SNAP header. The OUI value of 0x00-00-00 (3 octets) indicates that the following two bytes is an Ethertype. The Ethertype value of 0x08-06 (2 octets) indicates ARP. The total size of the LLC/SNAP header is fixed at 8 octets. This aligns the start of the ATMARP packet on a 64-bit boundary relative to the start of the AAL5 SDU.

6.1.8 IP Broadcast and Multicast Address

ATM does not support broadcast and multicast addressing, therefore, there are no mappings available from IP broadcast/multicast addresses to ATM addresses. This is also the main reason why IP RIP (route information protocol) is not supported on classic IP, while RIP extensively uses broadcasts to find out routes.

6.2 Using Classical IP on the MSS Server

Before starting to explain how the Classical IP functions can be configured on the IBM 8210 Nways MSS Server, we have to point out which methods are supported to establish client-to-client connections:

1. Using SVCs and an ARP server

   The 8210 is capable of participating in a LIS where an ARP server is used to enable LIS clients to learn the ATM address of other clients, and LIS-to-LIS connections are dynamically set up using SVCs.

   Using SVCs and an ARP server within your LIS minimizes the number of definitions required within each of the LIS clients and reduces the number of VCCs needed, as client-to-client connections are only established when needed.

   Note: On the 8210 an ARP server performs the role of both ARP server and client.

2. Using SVCs and no ARP server

   The 8210 is also capable of participating in a LIS where LIS-to-LIS client connections are statically defined, but dynamically set up using SVCs. No ARP server is used and because LIS clients cannot learn the ATM address of partner LIS clients, partner ATM addresses have to be predefined.

   Using SVCs without ARP server does not necessitate an ARP server and allows you to benefit from the dynamics within your ATM network. However, extra configuration effort is required and VCCs are immediately activated when the 8210 becomes active. SVCs without ARP server are typically used when the number of LIS clients is small.

3. PVCs and no ARP server

   The 8210 is capable of participating in a LIS where LIS-to-LIS client connections are statically set up using PVCs. Because all connections are predefined, no ARP server is required.

   When using PVCs, configuration in the adjacent ATM switches is required to define PVCs between partner LIS clients.

   Using PVCs does not necessitate an ARP server. However, extra configuration effort, including PVC definitions on your ATM switches, is required and VCCs are immediately activated when the 8210 becomes active. PVCs without an ARP server are typically used when the number of LIS clients is small and the ATM network does not provide SVC support.

Theoretically, although unlikely to occur, you can have a single LIS within which all three above methods of establishing LIS-to-LIS connections are used. Most likely, however, is that within a LIS only one of these methods is used.

6.3 Classical IP Configuration Overview

This section covers the basic steps required to configure MSS Server′s Classical IP functions using the Configuration Program.

Note: Definition scenarios using the command line interface are included within Chapter 11, “Implementation Scenarios” on page 289.

The different configuration steps that need to be completed depend on the functions that need to be activated. The basic steps include:

1 Define the ATM port(s)

ARP servers and LIS clients are components of the 8210 that all need to be associated with an ATM port. When associated, all ATM traffic of the component will take place on the ATM port specified.

During configuration of the ATM port you have to configure:

a. The ATM port attributes

   For configuration details see Chapter 3, “MSS Server and ATM Ports” on page 53.

b. The end system identifiers (ESIs) associated with the ATM port

   During the definition of the ARP server and/or clients either the burned-in end system identifier (ESI) or a user-defined ESI must be used. To ease the definition of the ATM addresses on remote LIS clients, for example, when referring to the ARP server, and to simplify troubleshooting it is recommended that you use a locally administered ESI. ESIs are administered per ATM port. A separate value has to be defined per ATM port. All LIS client/servers that connect using a particular port can use the same ESI. Make sure that when defining multiple 8210s, the user-defined ESIs are unique.

The ATM port configuration must be repeated for each ATM port on your MSS Server which is used for Classical IP functions.

2 Configure LIS Client

The configuration information that must be entered to define a LIS client is:

a. IP address

Make sure that all clients (including the ARP server) use a unique address within the range of IP addresses associated with the LIS. During the definition the IP address is associated with one of the MSS Server′s ATM interfaces. Use either interface 0 or interface 1, when you want all this LIS client traffic to use either ATM port 1 or 2, respectively (see the discussion in 7.1.1, "Interface Numbers" on page 174).

b. Subnet mask

Make sure that all clients within a logical IP subnet (LIS) use the same subnet mask.

c. End system identifier (ESI)

During the definition of the LIS client either the burned-in end system identifier (ESI) or a user-defined ESI must be used. Although, when using an ARP server, LIS client ATM addresses are learned dynamically and no predefinition of addresses is required, so we recommend the use of a locally administered ESI. All LIS clients using the same 8210 ATM port can use the same ESI.

d. Selector (SEL) byte

The 1-byte selector byte (SEL) decides, together with the 6-byte ESI and the 13-byte ATM network identifier, the ATM address used by the LIS client. For LIS clients using ARP services, it is recommended that you use an SEL byte generated by the MSS Server at run-time.

e. Service data unit (SDU)

Make sure that all clients within a logical IP subnet (LIS) use the same SDU. Use the default value (9188) whenever possible.

The maximum SDU size can be configured on a LIS client basis but cannot be greater than the maximum SDU size for the ATM interface (default 9234). Although the SDU can be defined on client basis, values are not independent because all clients on an ATM interface share the same MTU. The MTU size is set to the smallest client SDU size-8 (frames have 8 byte header). Consequently, all clients with a given ATM interface must have the same MTU, and therefore, care should be exercised when altering client′s maximum SDU size.

f. ARP server ATM address

When configuring a LIS client, the configurator requires you to specify if the LIS client is also an ARP server. If you want LIS clients functions only, configure NO.

To enable connectivity, LIS clients need to connect to an ARP server first. Make sure that the server ATM address specified is the ATM address of your ARP server.

The previous steps need to be repeated for every LIS client. For details see 6.3.1, “LIS Client Using Dynamic SVCs” on page 157.

3 Configure ARP Server

ARP servers defined on the MSS Server are both LIS client and ARP server, therefore, the definitions required to define an ARP server are to a large extent equivalent to defining a LIS client. The configuration steps required are:

a. IP address

   Make sure that all clients (including the ARP server) use a unique address within the range of IP addresses associated with the LIS. During the definition the IP address is associated with one of the MSS Server′s ATM interfaces. Use either interface 0 or interface 1, when you want all this ARP server traffic to use either ATM port 1 or 2, respectively.

b. Subnet mask

   Make sure that all clients/servers within a logical IP subnet (LIS) use the same subnet mask.

c. End system identifier (ESI)

   During the definition of an ARP server it is recommended that you use a user-defined ESI.

d. Selector (SEL) byte

   The 1-byte selector byte (SEL) decides, together with the 6-byte ESI and the 13-byte ATM network identifier, the ATM address used by the ARP server. Because the ARP server′s address needs to be hard-coded on LIS clients, it is recommended that you use a user-defined value, instead of using a value decided at run-time.

   Note: Make sure that when defining the selector byte, the combination of ESI and SEL on your IBM 8210 Nways MSS Server is unique.

e. SDU

   Make sure that all clients (including the ARP server) within a logical IP subnet (LIS) use the same SDU. Use the default value (9188) whenever possible. Note, see also the discussion of the LIS client′s SDU size.

f. ARP server ATM address

   When configuring a combined LIS client/ARP server, the configurator requires you to specify if the LIS client is also an ARP server. When defining an ARP server, configure YES.

The previous steps need to be repeated for every ARP server. For details see 6.3.2, “Configuring an ARP Server” on page 161.

4 Configure PVC connection

If some or all of your LIS clients do not support SVCs, PVCs can be used in addition to, or as an alternative for, the dynamic SVCs and/or static SVCs discussed in 2 on page 155 and in 5 on page 157, respectively.

6.3.3, "LIS Client Using PVCs" on page 162 discusses how to define a PVC between two LIS clients. Note that before defining the PVC a LIS client has to be configured.

Notes:

a. The PVC is a connection between two LIS clients. Equivalent definition are required at both ends.

b. Per LIS client you can define PVCs to multiple remote LIS clients.

5 Configure static SVC connection

If some or all of your LIS clients support SVCs but do not allow the use of an ARP server, static SVCs can be used in addition to, or as an alternative for, the dynamic SVCs and/or PVCs discussed in 2 on page 155 and in 4 on page 156, respectively.

6.3.4, "LIS Client Using Static SVCs" on page 164 discusses how to configure a predefined SVC between two LIS clients. Note that before being able to configure the SVC a LIS client has to be configured.

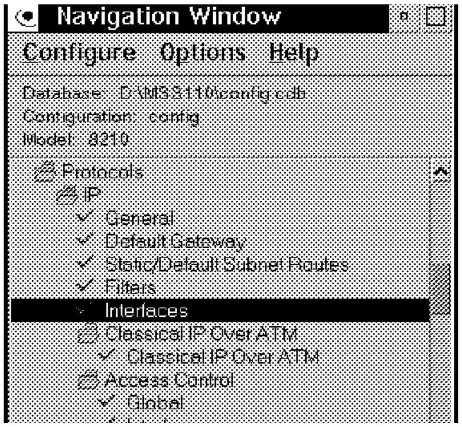
Notes:

a. The SVC is a connection between two LIS clients. Although both ends need to be configured as a LIS client, only one end needs to configure the SVC.

b. Per LIS client you can define static SVCs to multiple remote LIS clients.

6.3.1 LIS Client Using Dynamic SVCs

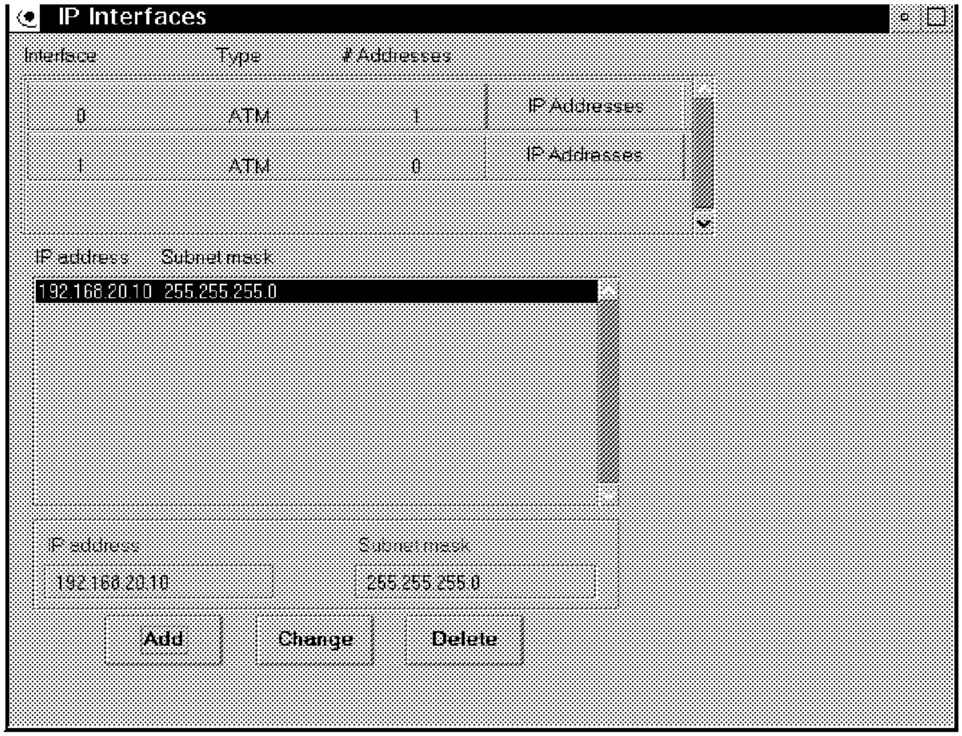
Figure 79 depicts the Navigation Window of the configuration program. Select Interfaces in the IP folder when you want to add a LIS client.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 79. Interfaces

As a result Figure 80 on page 158 appears. Click on IP Address for the interface which you want to use for this LIS client. Add a unique IP address that is consistent with the range of IP addresses associated with the LIS. Make sure that all LIS clients within the same LIS use the same subnet mask.



IP Interfaces

Interface	Type	# Addresses	
0	ATM	1	IP Addresses
1	ATM	0	IP Addresses

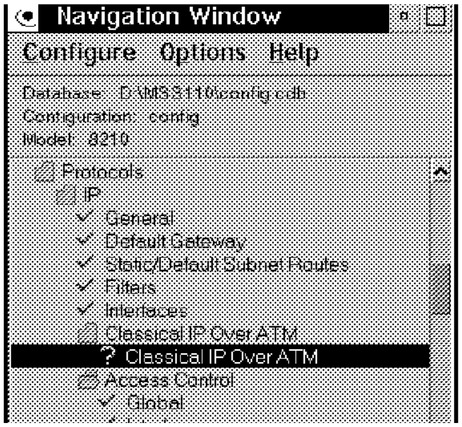
IP address	Subnet mask
192.168.20.10	255.255.255.0

IP address: 192.168.20.10

Subnet mask: 255.255.255.0

Figure 80. Add IP Address

Adding an IP address on interface 0 or 1 requires Classical IP over ATM definitions, as depicted in Figure 81.



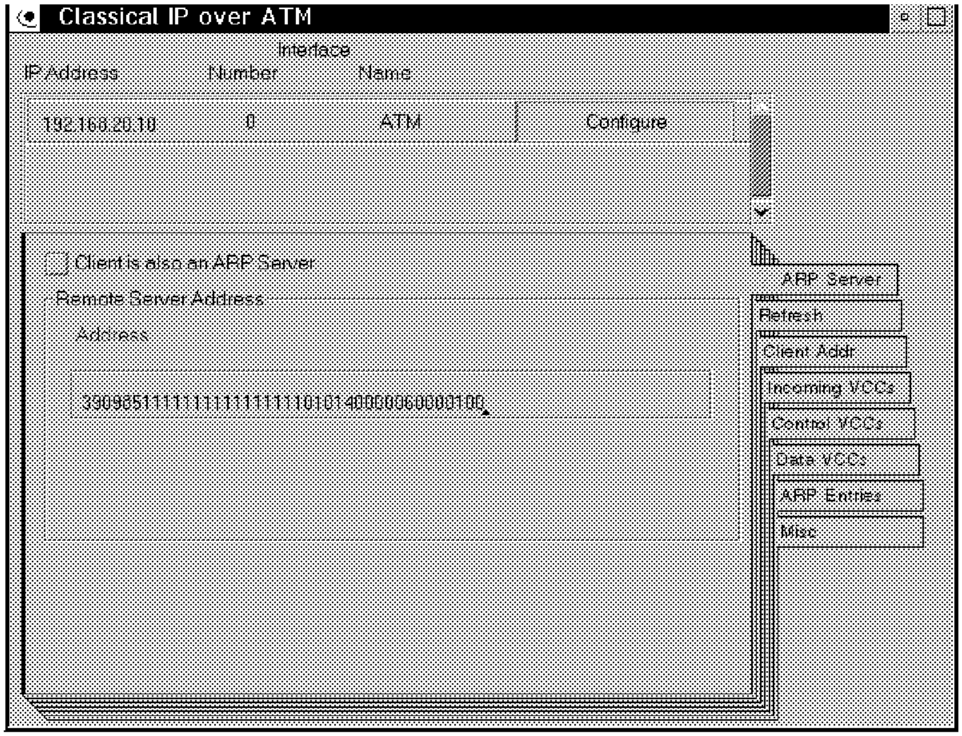
Navigation Window

Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

```
Protocols
  IP
    ✓ General
    ✓ Default Gateway
    ✓ Static/Default Subnet Routes
    ✓ Filters
    ✓ Interfaces
    Classical IP Over ATM
      ? Classical IP Over ATM
    Access Control
      ✓ Global
```

Figure 81. Classical IP over ATM Definitions

After clicking on Classical IP over ATM definitions, Figure 82 on page 159 appears. The configuration of the LIS client′s specific parameters can start after you have selected the proper IP address (the address defined in Figure 80).



IP Address	Interface Number	Interface Name
192.168.20.10	0	ATM

□ Client is also an ARP Server

Remote Server Address

Address

3909851111111111111111010140000060000100

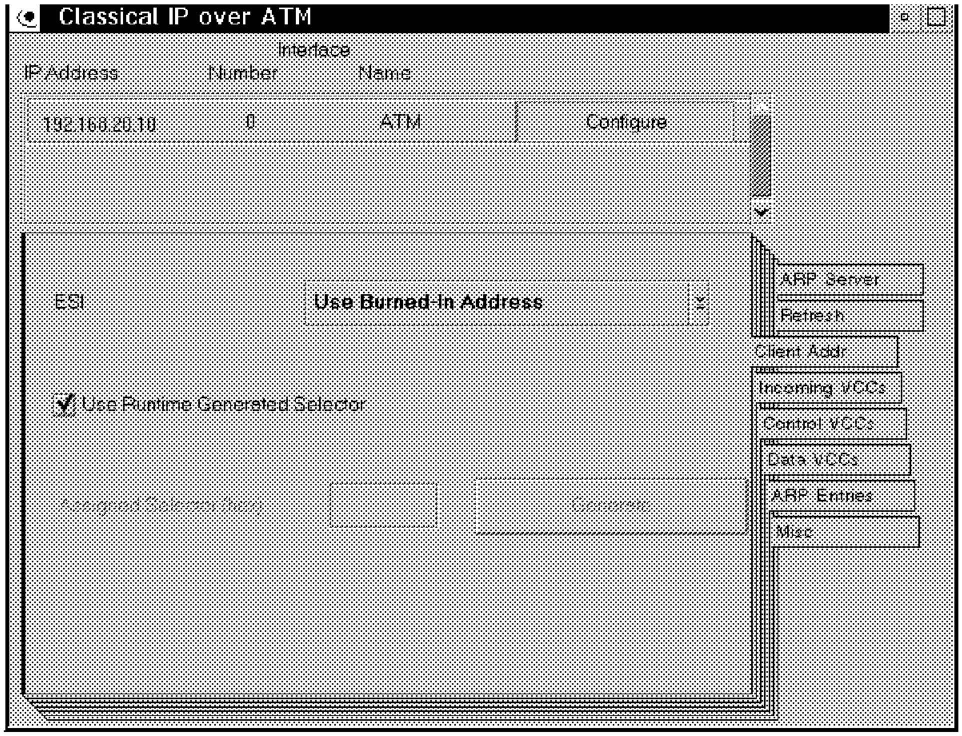
Figure 82. ARP Server

When defining a LIS client do not enable Client is also an ARP Server. Instead, enter the 20-byte ATM address of the ARP server.

This, using default values in the remaining configurator screens, completes the LIS client configuration. Two screens, however, are worth mentioning.

Figure 83 on page 160 shows that for a LIS client the configurator assumes that the selector is assigned at run-time. This setting is adequate, unless using predefined SVCs between two clients (see 6.3.4, “LIS Client Using Static SVCs” on page 164). In this case you have to make sure that on at least one of the clients, a preconfigured selector is used.

Figure 84 on page 160 shows the maximum SDU (service data unit) that is used by the LIS clients. Make sure that all LIS clients within the same LIS use the same value. Also make sure that this value is less than the maximum size allowed on the ATM port (see the discussion of *max-frame* in Chapter 3, “MSS Server and ATM Ports” on page 53).



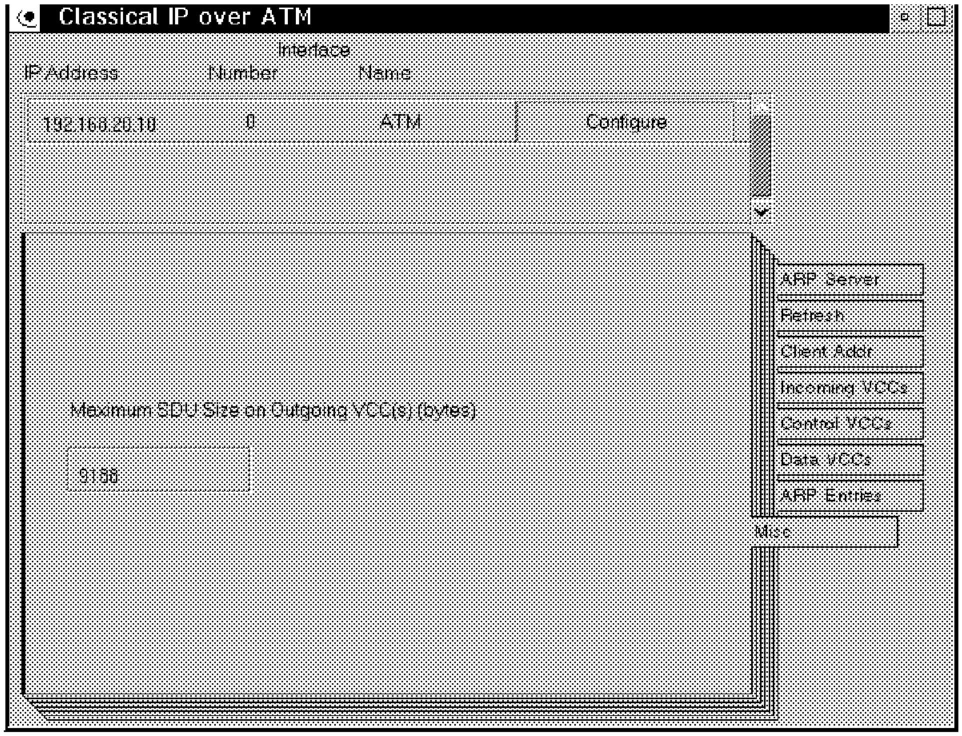
IP Address	Interface Number	Name
192.168.20.10	0	ATM

ESI: Use Burned-In Address

☑ Use Runtime Generated Selector

Assigned Selector (hex)

Figure 83. Run-Time Selector



IP Address	Interface Number	Name
192.168.20.10	0	ATM

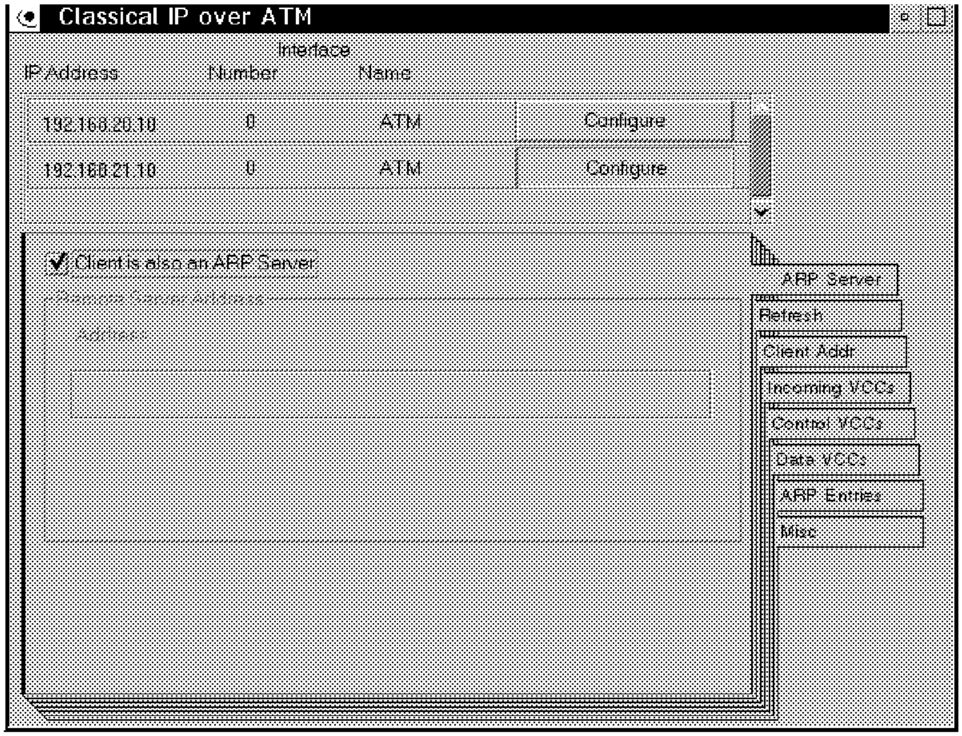
Maximum SDU Size on Outgoing VCC(s) (bytes): 9188

Figure 84. Maximum SDU Size

6.3.2 Configuring an ARP Server

ARP server and LIS client (see 6.3.1, “LIS Client Using Dynamic SVCs” on page 157) configurations are defined using the same configuration screens. Also for an ARP server you have to define an IP address (+subnet mask) first and associate this address with a specific ATM port (see Figure 79 on page 157, Figure 80 on page 158, and Figure 81 on page 158).

Figure 85 appears when clicking on Configure for the newly added IP address.



Classical IP over ATM

IP Address	Interface Number	Interface Name	
192.168.20.10	0	ATM	Configure
192.168.21.10	0	ATM	Configure

☑ Client is also an ARP Server

Remote Server Address

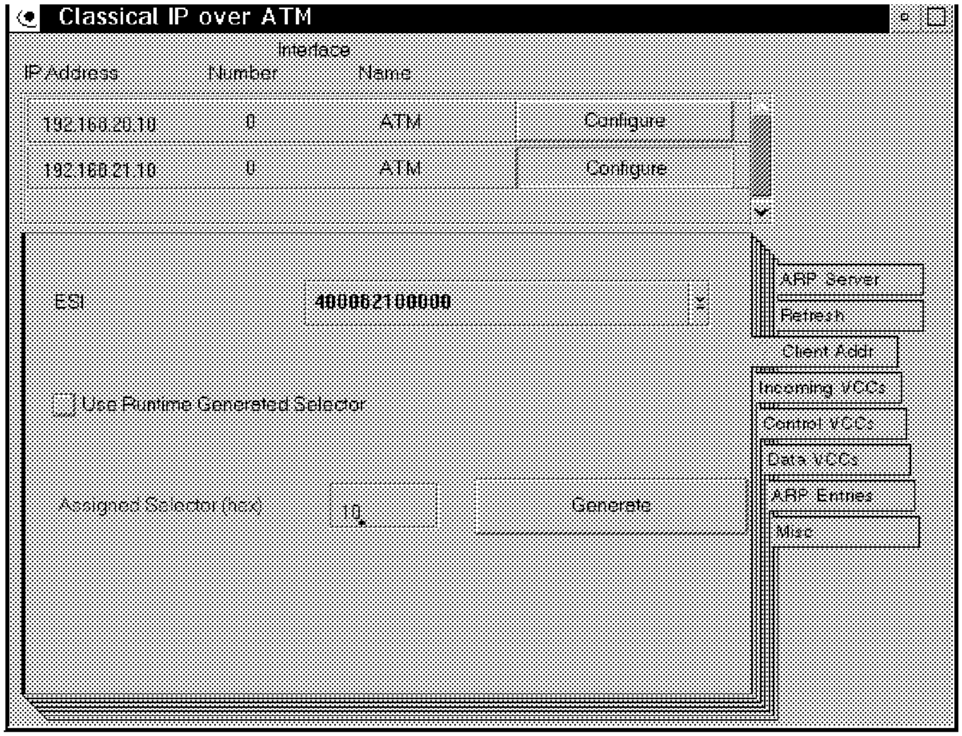
Address

Figure 85. ARP Server Definition

When defining an ARP server enable Client is also an ARP Server. Leave the Remote Server Address field empty.

One configuration screen that needs special attention is Figure 86 on page 162. This screen becomes available after selecting Client Addr. As the ARP server′s ATM address need to be specified during the configuration of remote LIS clients, the variables that comprise the address (in particular ESI and SEL) need to be fixed. It is, therefore, advised that you use a locally administered ESI and a user-defined selector.

For the remainder of the configuration screens, default values can be used. Make sure (see, for example, Figure 84 on page 160) that the maximum SDU size does not conflict with the value configured on other LIS clients.



Classical IP over ATM

IP Address	Interface Number	Name	
192.168.20.10	0	ATM	Configure
192.168.21.10	0	ATM	Configure

ESI 400062100000

□ Use Runtime Generated Selector

Assigned Selector (hex) 10

Figure 86. ESI and Selector

6.3.3 LIS Client Using PVCs

When defining a LIS client you have the option to provide connectivity using SVCs or PVCs. The use of SVCs is more flexible and is recommended. However, in situations where your ATM switches do not support SVCs, or no ARP server is available for your LIS, PVC connections could be considered. PVCs can also be considered if UNI incompatibilities exists.

Note: PVCs can be used in conjunction with SVCs as well.

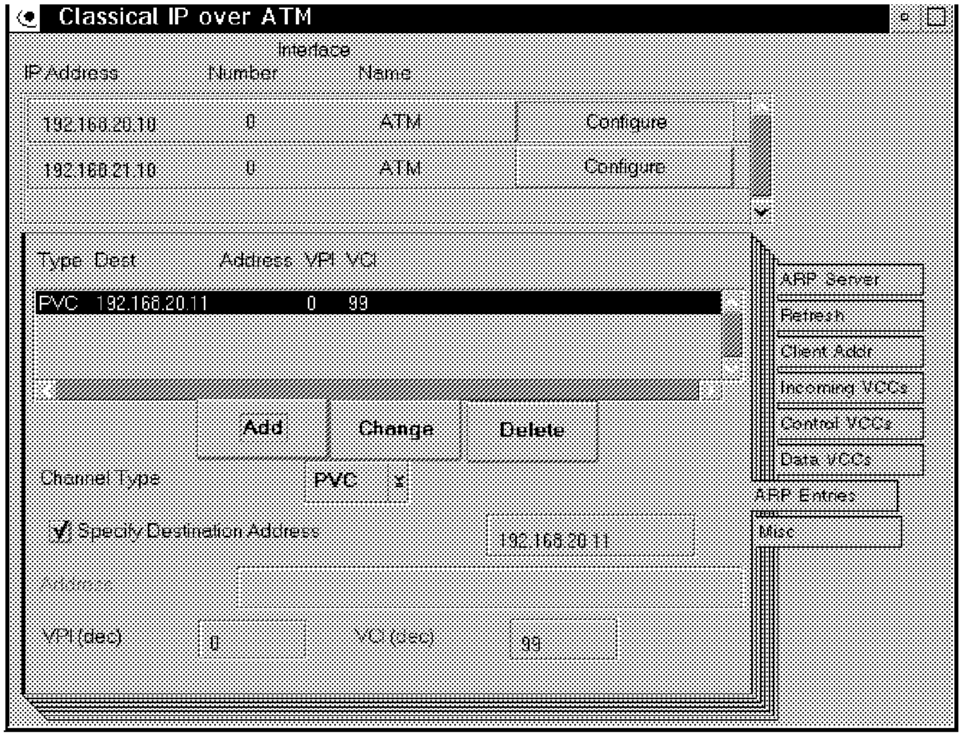
To define a PVC to a remote LIS client requires two things:

- Define LIS client
- Define PVC definition

The LIS client definitions have been overviewed in 6.3.1, “LIS Client Using Dynamic SVCs” on page 157 and are not repeated. They are required at both ends of the PVC. The LIS client definitions define the throughput characteristics of the PVC.

Also the PVC definition requires similar configuration at both ends. PVC definitions are entered during the definition of the LIS client. Per LIS client (identified by its IP address) you can define multiple PVCs.

The parameters that can be entered become available after selecting ARP Entries during the configuration of Classical IP over ATM definitions. Figure 87 on page 163 appears.



Classical IP over ATM

IP Address	Interface Number	Name	
192.168.20.10	0	ATM	Configure
192.168.21.10	0	ATM	Configure

Type	Dest Address	VPI	VCI
PVC	192.168.20.11	0	99

Channel Type PVC

☑ Specify Destination Address 192.168.20.11

Address

VPI (dec) 0 VCI (dec) 99

Figure 87. PVC Definition

Make sure that the virtual path identifier (VPI) and virtual channel identifier (VCI) match the definitions on the adjacent switch (see 6.3.3.1, "Defining PVCs on the ATM Switch"). When Specify Destination Address has been enabled, the IP address of the remote LIS client needs to be entered. It is recommended that you disable this option and let the MSS Server learn the IP address of the other end dynamically. This, however, requires InATMARP support at the other end. InATMARP is supported on the MSS Server.

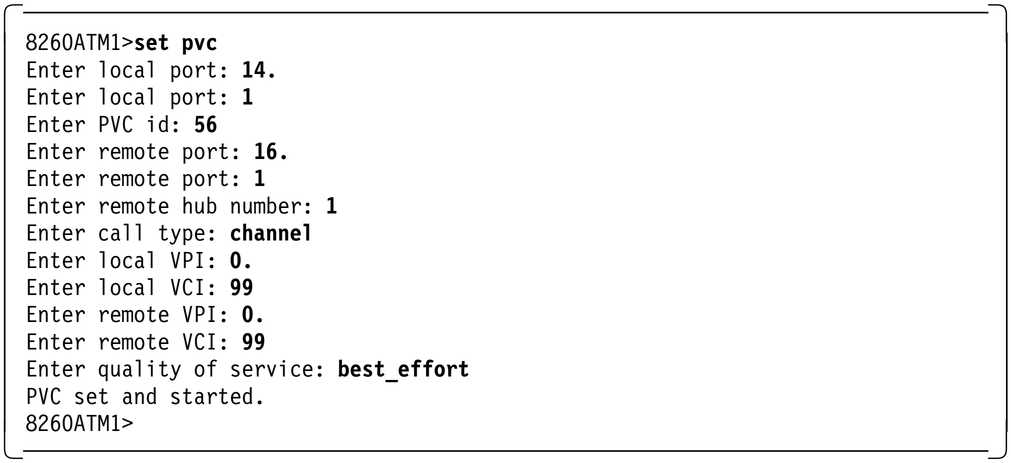
6.3.3.1 Defining PVCs on the ATM Switch

Figure 88 on page 164 shows the `set pvc` command to define a PVC between two locally attached devices, port 14.1 and 16.1, respectively. The PVC is verified using the `show pvc` command.

On both ends of the PVC, we have used VPI number 0 and VCI number 99. Both endpoints of our PVC attach to ATM hub number 1. Replace these with values that are relevant for your configuration.

Note: The VPI/VCI on both ends of the PVC can be different as well.

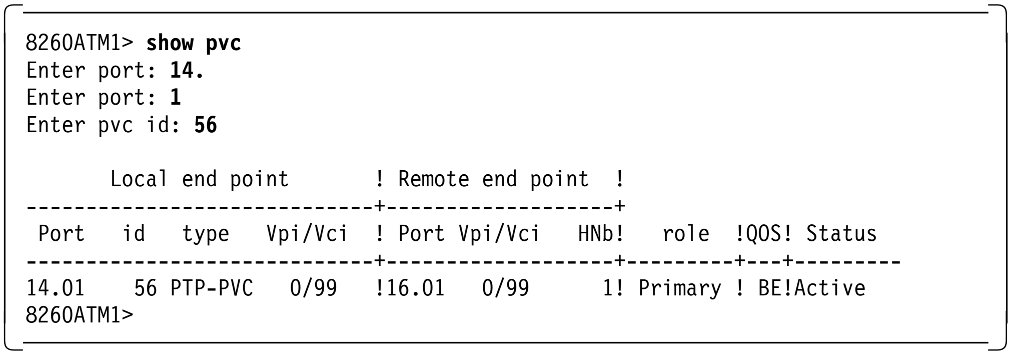
Define PVC:



```
8260ATM1>set pvc
Enter local port: 14.
Enter local port: 1
Enter PVC id: 56
Enter remote port: 16.
Enter remote port: 1
Enter remote hub number: 1
Enter call type: channel
Enter local VPI: 0.
Enter local VCI: 99
Enter remote VPI: 0.
Enter remote VCI: 99
Enter quality of service: best_effort
PVC set and started.
8260ATM1>
```

Figure 88 (Part 1 of 2). PVC Definitions on ATM Switch

Verify PVC definition:



```
8260ATM1> show pvc
Enter port: 14.
Enter port: 1
Enter pvc id: 56

       Local end point       ! Remote end point  !
-----------------------------+-------------------+
 Port   id   type   Vpi/Vci  ! Port Vpi/Vci   HNb!   role  !QOS! Status
-----------------------------+-------------------+---------+---+---------
14.01    56 PTP-PVC   0/99   !16.01   0/99      1! Primary ! BE!Active
8260ATM1>
```

Figure 88 (Part 2 of 2). PVC Definitions on ATM Switch

Note: PVC identifier 56 has been assigned to the PVC definition.

6.3.4 LIS Client Using Static SVCs

The IBM 8210 Nways MSS Server provides an interesting option of being able to configure LIS clients that are using SVCs for their LIS-to-LIS client connections, but do not require the presence of an ARP server. Similar to using PVCs, this approach has the advantage that no ARP server is needed. In addition, because the LIS-to-LIS client connections are established using SVCs, no ATM switch definitions are required to enable the VCCs.

Note: Static SVCs can be used in conjunction with dynamic SVCs and PVCs.

6.3.4.1 Active and Passive LIS Client

Predefined SVCs require that for each client-to-client connection one client′s ATM address is defined on the partner client. This results in definitions that are not symmetrical; one (active) client that defines the ATM address of the other end and is responsible for VCC establishment, and one (passive) client that awaits VCC establishment. After the connection has been established, full-duplex IP transport between both clients is possible.

Note: If the SVC has been define on both ends, depending on timing, two VCCs can be established which are each used for traffic in one direction.

6.3.4.2 Definitions Required

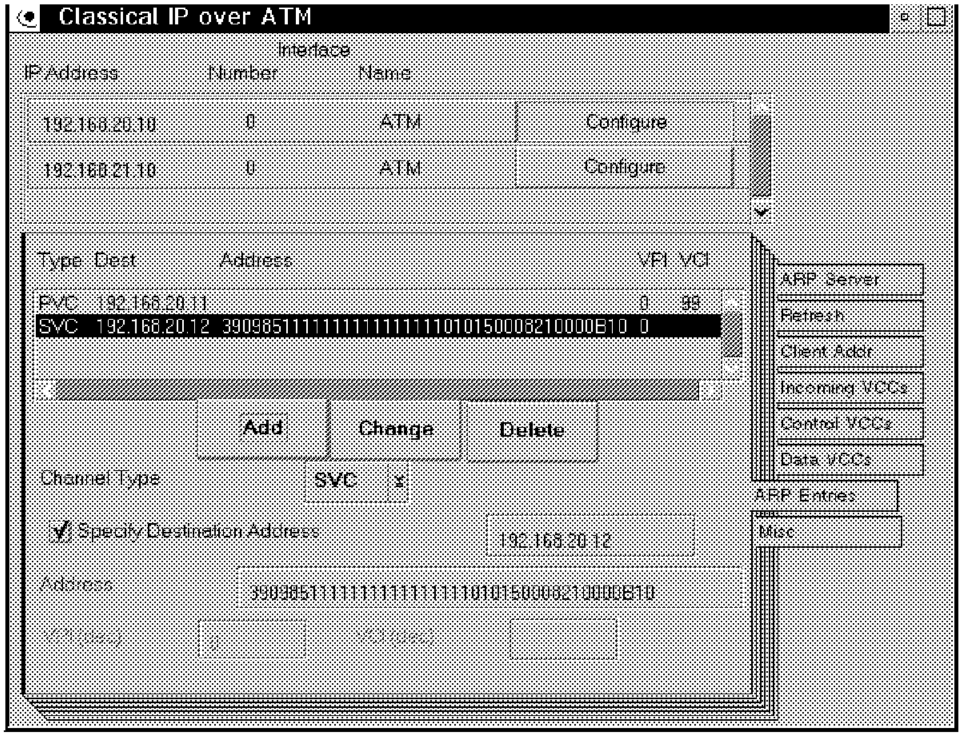
To define a static SVC to a remote LIS client requires two things:

- Define LIS client
- Define SVC definition

The LIS client definitions have been overviewed in 6.3.1, “LIS Client Using Dynamic SVCs” on page 157 and are not repeated. They are required at both ends of the SVC. The LIS client definitions define the throughput characteristics of the PVC.

The SVC definitions need to be entered on one end only. This end is referred to as the active LIS client, because it is responsible for VCC establishment. Definitions are added during the definition of the LIS client. Per LIS client (identified by its IP address) you can define multiple static SVCs.

The parameters that can be entered become available after selecting ARP Entries during the configuration of Classical IP over ATM definitions. Figure 89 results.



Classical IP over ATM

IP Address	Interface Number	Interface Name	
192.168.20.10	0	ATM	Configure
192.168.21.10	0	ATM	Configure

Type	Dest	Address	VPI	VCI
PVC	192.168.20.11		0	99
SVC	192.168.20.12	3909851111111111111111010150008210000B10	0	

Channel Type: SVC

☑ Specify Destination Address: 192.168.20.12

Address: 3909851111111111111111010150008210000B10

VPI (dec): 0

VCI (dec):

Figure 89. Static SVC Definition

When Specify Destination Address has been enabled, the IP address of the remote LIS client needs to be entered. It is recommended that you disable this option and let the MSS Server learn the IP address of the other end dynamically. This, however, requires InATMARP support at the other end. InATMARP is supported on the MSS Server.

The Address field must match the ATM address of the remote LIS client. To make sure that the ATM address of the passive LIS client is fixed, it is recommended that you specify a locally administered ESI and a preconfigured

Selector during its configuration. During the definition of the active LIS client you can specify the use of a run-time Selector (compare Figure 86 on page 162 with Figure 83 on page 160).

Using the point-to-point (PtP) concept, more complex network structures can be built. Hereby you need to be aware that within each LIS:

- Every LIS client only requires a single IP address and a single LIS client definition.
- For each PtP connection, at least one end must be assigned as the active client. This active client is responsible for VCC establishment and requires an ARP entry.
- Clients can be active for one PtP connection, while being passive for another.

6.3.5 Redundant ARP Server

The ARP server is a critical component in the logical IP subnet and its availability must be maximized. When your network is using multiple 8210s, its ARP server redundancy feature can be used to increase client-to-client connectivity.

Figure 90 depicts two 8210s on which an ARP server has been defined with the same ATM address defined on both systems. Either 8210 can perform ARP server functions for the same LIS.

Note: It is advised that primary and backup ARP server use the same IP address as well.

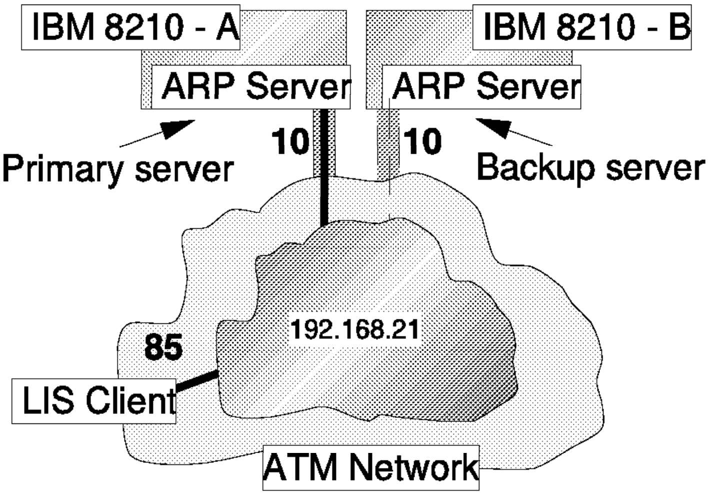


Figure 90. ARP Server Redundancy

When the remote LIS clients need to learn ATM addresses associated with other LIS clients, they require service of the ARP server. The ATM address of the ARP server is hard-coded on each LIS client.

The ARP server redundancy feature is based on the fact that the ARP server on both 8210s use the same ATM address, and duplicate ESI registration is retried.

Important

Using the same ATM address requires that both 8210s attach to the same ATM switch and have the same ESI and SEL defined. For details on defining ESI and LES, see 6.3.2, “Configuring an ARP Server” on page 161.

Because ATM switches do not allow an ESI to register twice, ESI registration of the backup 8210 will fail initially. But because 8210 has implemented an ESI registration retry mechanism, it will be successful when the primary ARP server fails.

After the backup ARP server has registered the ESI associated with the ARP server, LIS clients will re-establish their control VCC with the ARP server, and LIS communication can continue.

Important

Although the ARP server can be active on a single 8210 only, other functions using different ESIs can be active on both 8210s at the same time.

It is important to realize that the MSS Server that registers its ESI first becomes the active server. You cannot (yet), other than for the redundant LES/BUS (see 5.4, “Redundant LES/BUS” on page 106), predefine which server is primary or backup, respectively.

Chapter 7. MSS Server and IP Routing Protocols

This chapter provides hints and tips to designers of IP internetworks to allow them to make the best use of the IBM 8210.

It provides general information about the limitations of each protocol and when they should and should not be used. It also provides specific information to assist network designers in configuring each of the IP routing protocols on the IBM 8210.

Many networks demand the use of multiple interior gateway protocols or the use of interior and exterior gateway protocols in combination. When multiple protocols are used it is necessary to *export* routes from one protocol to another. The IBM 8210 route export capabilities are discussed in 7.6, "Routing Protocols Interoperability" on page 197.

7.1 Configuration Overview

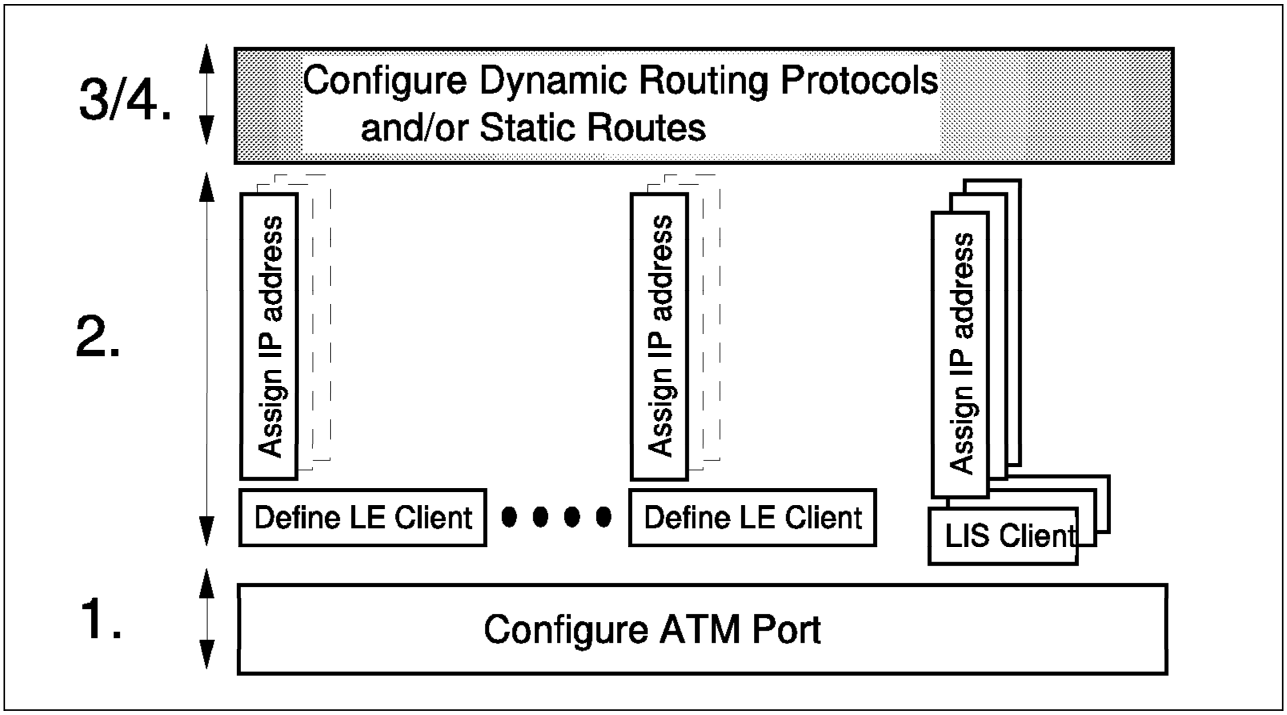
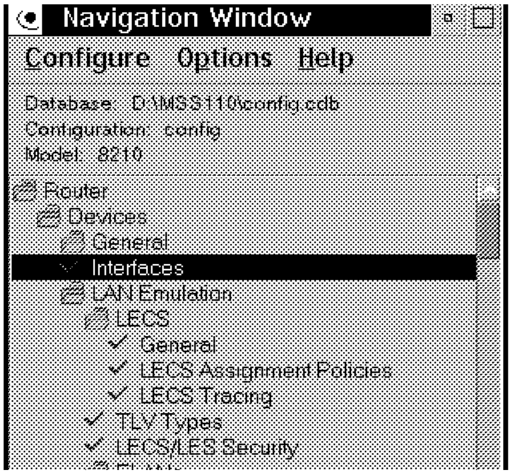


Figure 91. Configuring IP Functions

To enable the IBM 8210 Nways MSS Server for IP routing is a layered process (see Figure 91) that requires the following configuration steps:

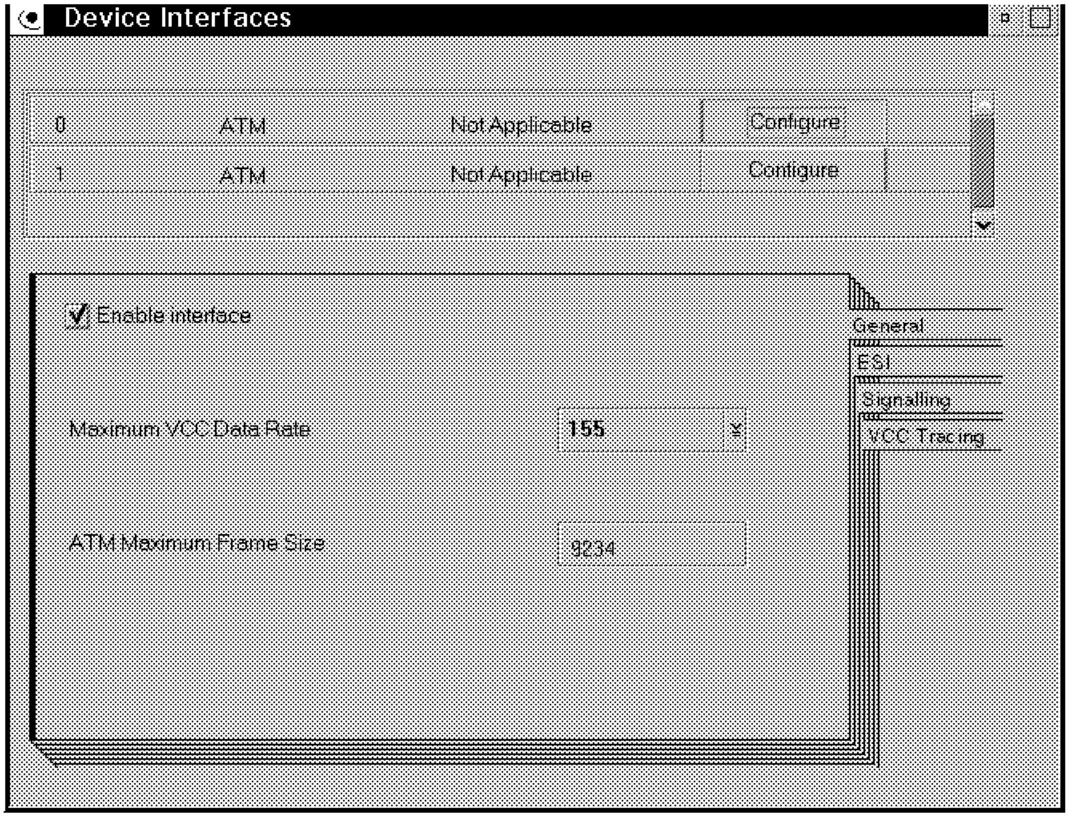
1 Configure ATM port

After an ATM port has been defined (see Figure 92 on page 170 and Figure 93 on page 170), logical IP subnet (LIS) and/or LAN emulation client(s) can be assigned to it.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 92. Interface Definition



0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

☑ Enable interface

Maximum VCC Data Rate: 155

ATM Maximum Frame Size: 9234

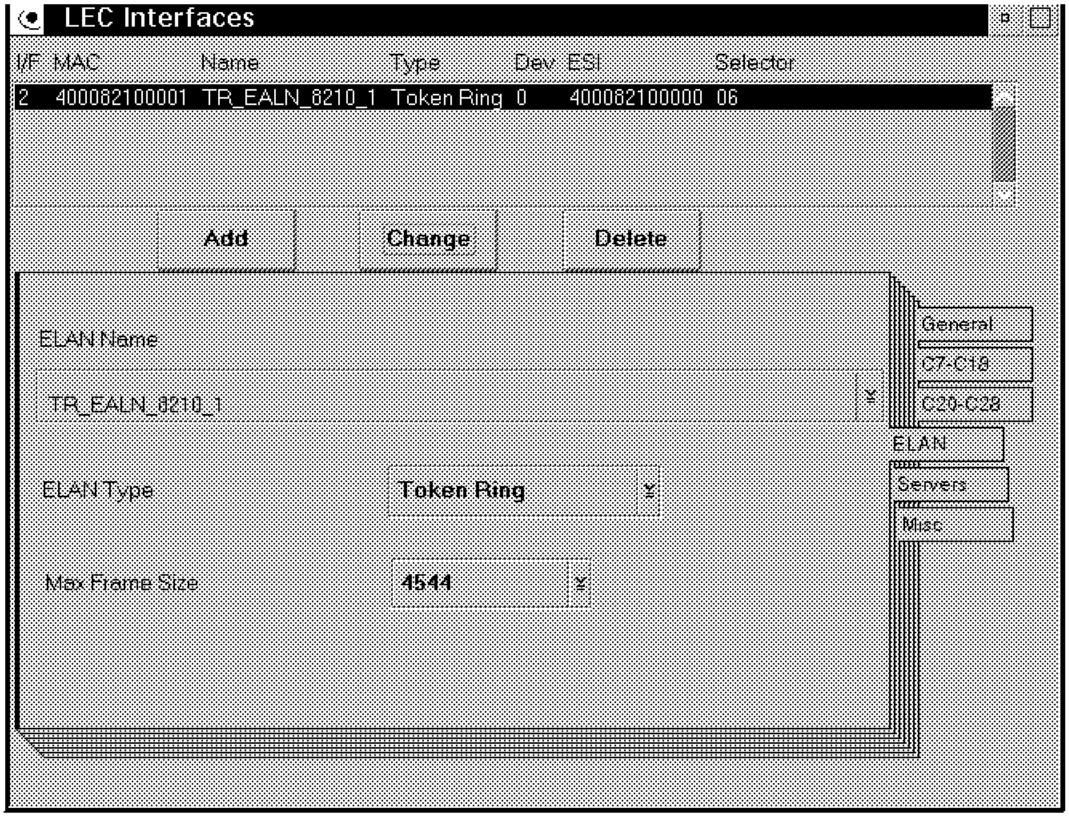
Figure 93. Enable ATM Port

2 Define IP interface(s)

To enable IP routing requires that one or multiple IP addresses have been assigned to the IBM 8210 Nways MSS Server. The procedure for adding an IP address depends on whether you use IP over LAN emulation or Classical IP.

a. LAN Emulation

IP over LAN emulation requires that you define an LE client (see Figure 94) first. Defining the LE client results in a logical interface to which you then assign an IP address (Figure 95 on page 172).



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100001	TR_EALN_8210_1	Token Ring	0	400082100000	06

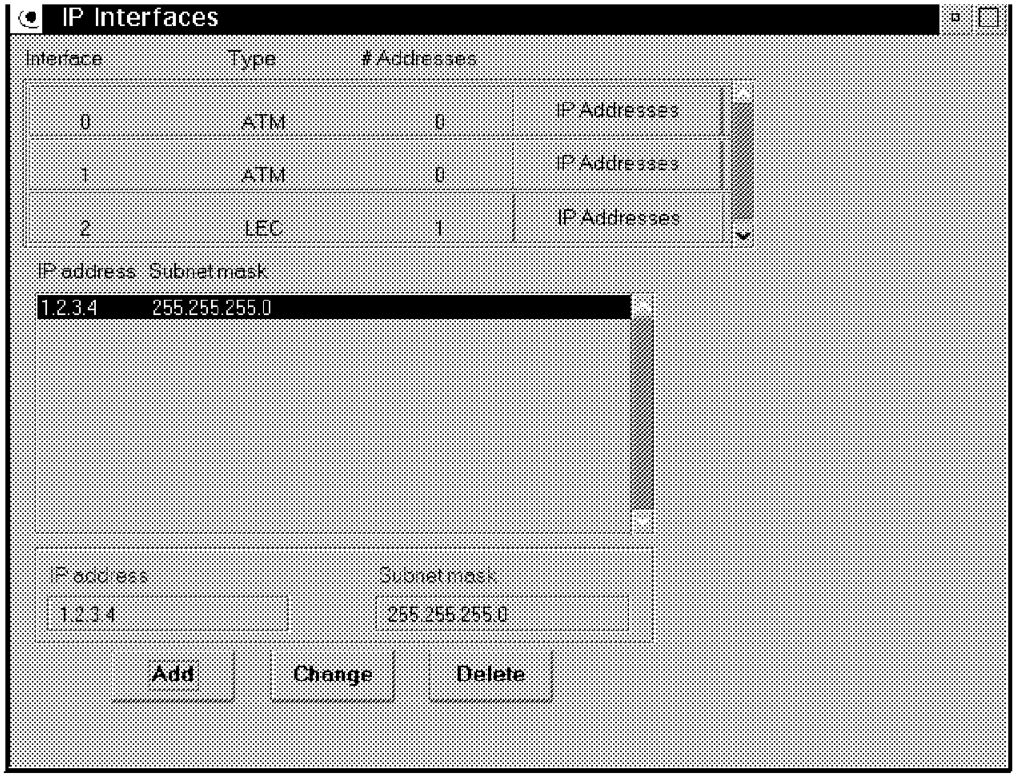
ELAN Name

TR_EALN_8210_1

ELAN Type: Token Ring

Max Frame Size: 4544

Figure 94. Definition of LAN Emulation Client



Interface	Type	# Addresses	
0	ATM	0	IP Addresses
1	ATM	0	IP Addresses
2	LEC	1	IP Addresses

IP address	Subnet mask
1.2.3.4	255.255.255.0

IP address: 1.2.3.4
Subnet mask: 255.255.255.0

Figure 95. Adding an IP Address to the LAN Emulation Client

Important

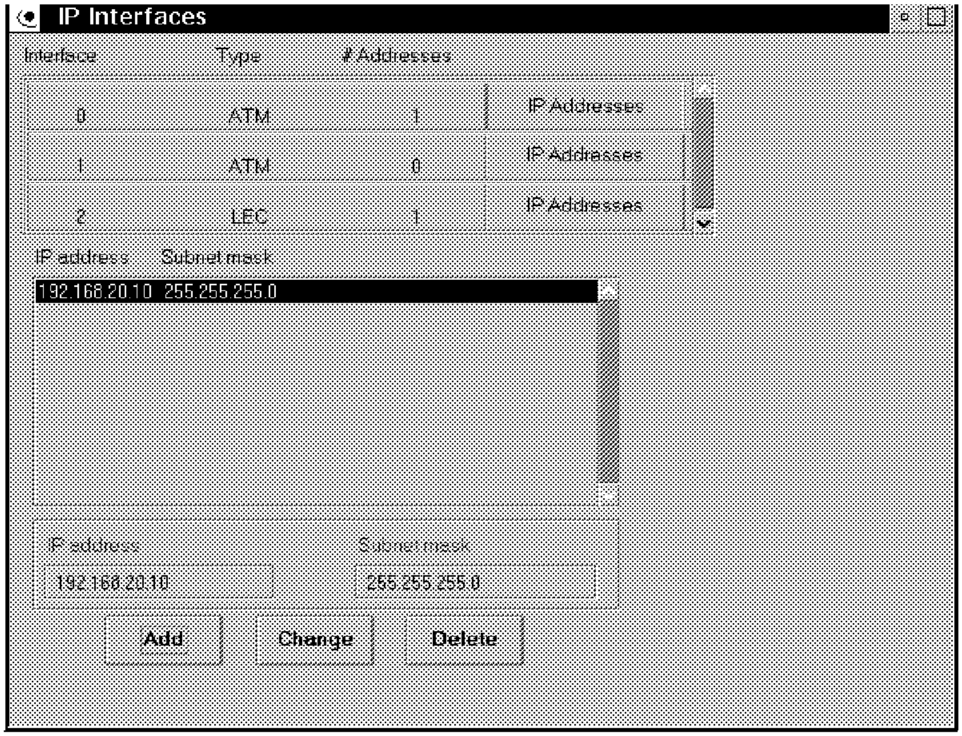
During the definition of your LE client, a logical interface number (I/F in Figure 94 on page 171) is generated. This number must be used when defining the IP address.

b. Classical IP

When using Classical IP you first assign (Figure 96 on page 173) the IP address, whereby the IP address is either assigned to interface 0 or *n*. Hereby interface 0 and *n* correspond to all Classical IP addresses defined on ATM port 1 and 2 respectively. After the IP address has been defined, the associated LIS client and/or server must be defined (see Figure 97 on page 173).

Note: *n* is a variable that depends on the order in which definitions have been entered (see 7.1.1, “Interface Numbers” on page 174).

Note: For SNMP management and TELNET and/or HTML configuration and operation services, you can use any of the IP addresses defined.



IP Interfaces

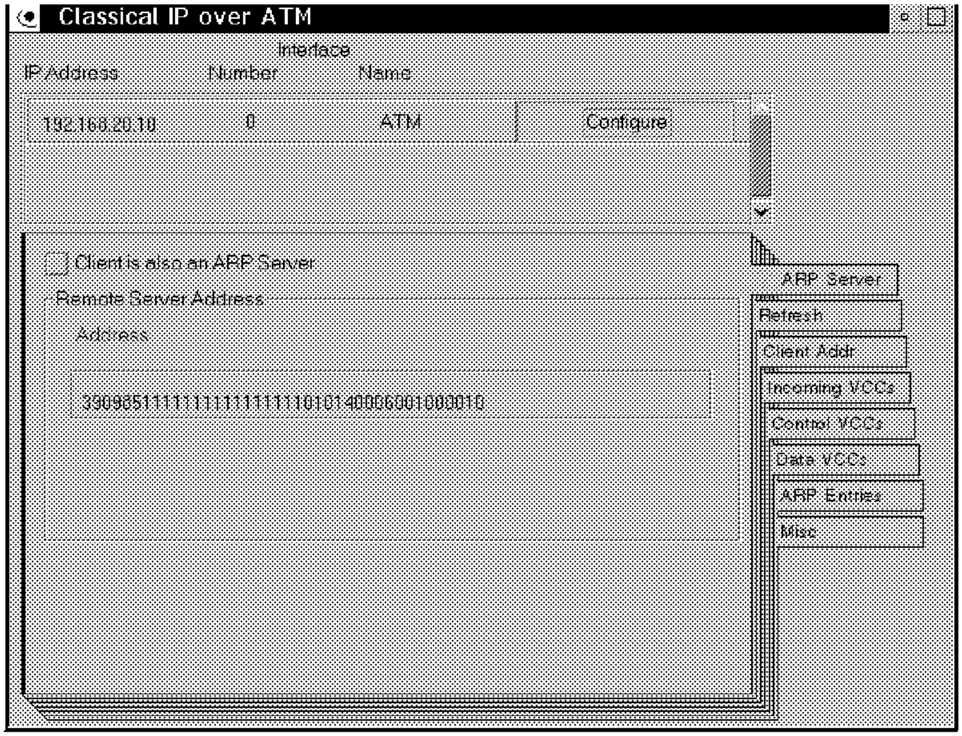
Interface	Type	# Addresses	
0	ATM	1	IP Addresses
1	ATM	0	IP Addresses
2	LEC	1	IP Addresses

IP address	Subnet mask
192.168.20.10	255.255.255.0

IP address: 192.168.20.10

Subnet mask: 255.255.255.0

Figure 96. Adding a LIS IP Address



Classical IP over ATM

IP Address	Interface Number	Interface Name	
192.168.20.10	0	ATM	Configure

□ Client is also an ARP Server

Remote Server Address

Address

3909851111111111111111010140006001000010

Figure 97. Definition of a LIS Client and/or Server

3 Configure dynamic routing protocols

When more than a single IP address has been defined, the IBM 8210 Nways MSS Server becomes effectively an IP router. To enable the IBM 8210 Nways MSS Server to participate in the exchange of dynamic routing information throughout your network, you have to:

- Configure RIP, for details see 7.2, "Using RIP" on page 175
- Configure OSPF, for details see 7.3, "Using OSPF" on page 180
- Configure BGP, for details see 7.4, "Using BGP Version 4" on page 190

One or more of the dynamic IP routing protocols supported by the IBM 8210 Nways MSS Server can be enabled.

4 Configure static routes

As an alternative to the dynamic routing protocols, static routes can be specified. Static routes are usually adequate in simple (IP) network constructs, for example, when the IBM 8210 Nways MSS Server is not used for IP routing, and IP is merely used for management and operation services.

It is obvious that at least one of the ATM ports has to be configured to enable connectivity to your ATM network. The definition of IP address(es) is optional. A minimum of one IP address is required to enable operation and management access to the IBM 8210 Nways MSS Server via its ATM network (that is, *in-band*).

Notes:

1. When your 8210 is not employed in any IP-related functions, you can assign an IP address to it, using the IP host services.
2. The configuration and management functions can also be accessed via the SLIP interface.

At least two IP addresses are required to enable IP routing. The configuration of the dynamic routing protocols and/or static routes is optional, depending on the complexity of your IP network, and if the IBM 8210 Nways MSS Server is used for IP routing.

7.1.1 Interface Numbers

For its bridging and routing functions the IBM 8210 Nways MSS Server uses the concept of interfaces which are the ATM equivalent of network interfaces in a legacy network. The Classical IP support is similar to the support for conventional IP routers on X.25 or frame-relay networks while the ELAN routing functions are essentially the same as used by convential IP routers on legacy LANs.

Each interface contains an interface number. Interface numbers are unique per ATM port. The Configuration Program maps all LIS clients on to interface 0 for the first ATM port. The interface number that is used for LIS clients on the second ATM port depends on the order in which definitions have been added. If the ATM port is present at the time a new configuration is started, interface 1 will be used.

Interface Number

The command line configurator maps all LIS clients on ATM port 1 to interface 0. The interface number used for the the second ATM port is configuration dependent.

Each LE client definition results in a separate, unique (logical) interface number which must be remembered and used when an IP address is assigned to the LE client.

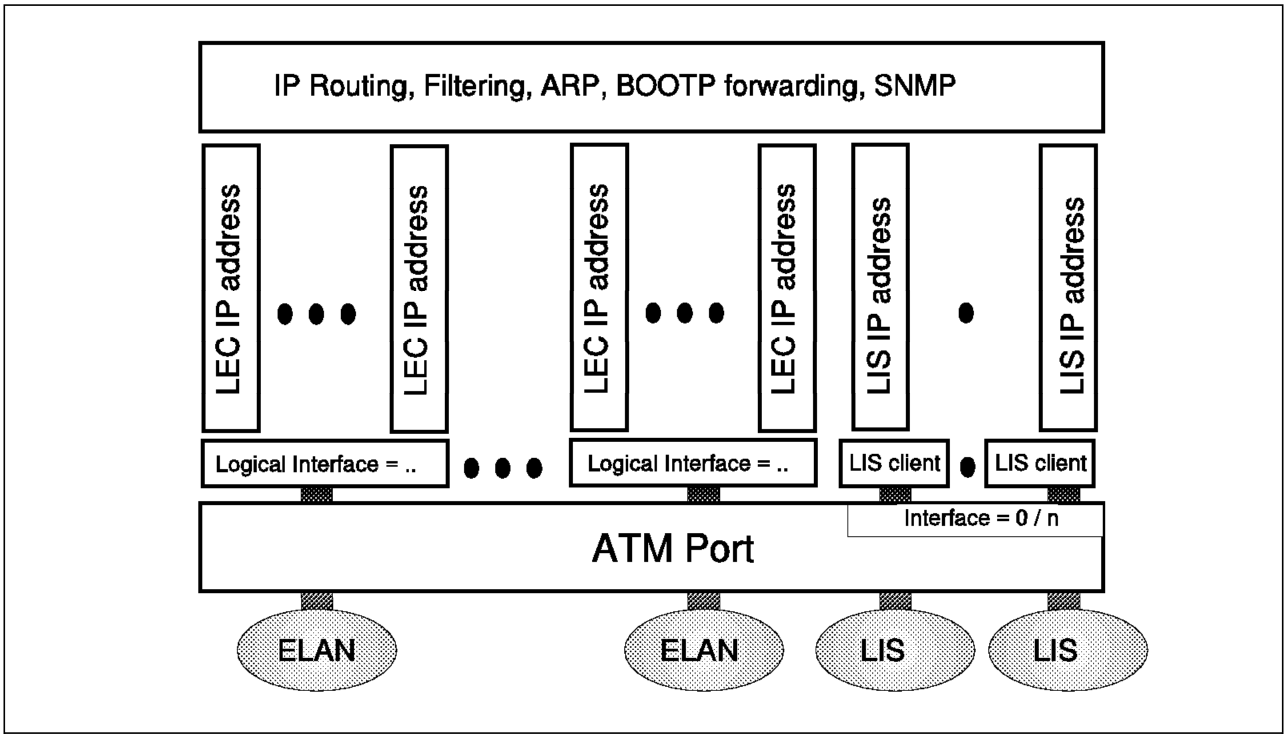


Figure 98. Interface Numbers

The IBM 8210 Nways MSS Server imposes a restriction of 32 IP addresses per interface. Therefore, as all LIS clients are associated with a single interface, you can define up to 32 LIS clients per ATM port. The maximum number of LAN emulation clients per IBM 8210 Nways MSS Server is 252, resulting in a virtually unlimited (32*252) number of IP addresses using LAN emulation.

Note: Be aware that the actual number of IP addresses depends on traffic patterns and the number of VCCs supported on the adjacent ATM switch.

7.2 Using RIP

RIP is currently the most common IP routing protocol. This is because it has been available with the UNIX operating system for some time and, because it is straightforward to implement, it has been ported to or re-implemented on other systems. It is supported by all major router vendors.

Most RIP networks are relatively small, and many have evolved to meet purely local requirements. In these networks general purpose computer systems are often used as convenience routers, and RIP has been used because such systems generally only support that single routing protocol.

There are major limitations to RIP. These limitations, expectedly, are highlighted when large internetworks are to be designed. They are discussed in B.3, “Routing Algorithms” on page 479 but are listed below for completeness:

- No RIP support on Classical IP subnet

> **LAN Emulation Only**
>
> RIP is not supported on Classical IP subnets due to its lack of broadcast support. Use OSPF, BGP, or static routes instead.

- No support for variable length subnet masks
- Limit of 15 hops in a network
- No support for alternate routes based on IP type of service
- Slow convergence on large internetworks (unless triggered updates supported, as is the case in the IBM 8210)
- Routing tables broadcast even if unchanged
- Routing tables can be very large as they contain all routes
- No support for discontiguous networks

  Note: A discontiguous network is a (class A, B, or C) network split up into multiple parts by one or more other networks.

RIP is a very robust protocol and there is a very high probability that implementations from different vendors will interwork without difficulty, but due to the previous limitations, it is not recommended that you use RIP as the major routing protocol for new network designs.

RIP should only be used in situations where it is necessary to interwork with an existing network already using RIP. If an existing internetwork is growing in size, it is recommended, where possible, that the extensions be implemented using OSPF. This can be done by migrating the entire network to OSPF or by running parts of the network with OSPF and parts with RIP. If this is not possible, an OSPF backbone could be implemented to link together the existing RIP-based networks.

If RIP is used in a network, the designer should be aware of its limitations and attempt to configure routers to reduce their impact to the minimum.

In order to optimize the use of RIP there are several guidelines that should be adopted when interface settings are configured. Their effect will be to minimize unnecessary network traffic and minimize RIP processing overhead.

- Do not enable RIP unless there is at least one interface on which it is used.
- Enable an IP interface for RIP only when there are RIP partners reachable via that interface.

7.2.1 RIP Configuration on the IBM 8210

Configuring the 8210 RIP functions requires the configuration of box and interface specific parameters (Figure 99 on page 177).

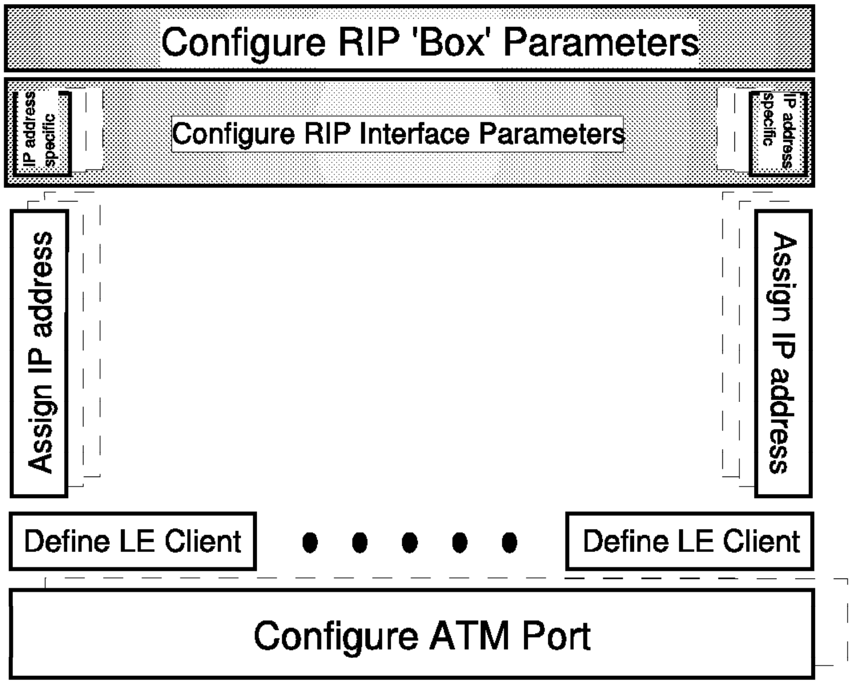


Figure 99. Configuring RIP Functions

Assuming the appropriate IP addresses have been defined, the Configuration Program requires the following steps (see Figure 115 on page 192) to complete the RIP configuration:

1 Enable RIP on box level

RIP is, by default, disabled and needs to be enabled before it can be used. This configuration option, see Figure 101 on page 178, becomes available after clicking on RIP General in the Navigation Window.

2 Specify routes RIP always accepts

The acceptance of routes RIP receives from neighbor routers depends on two things:

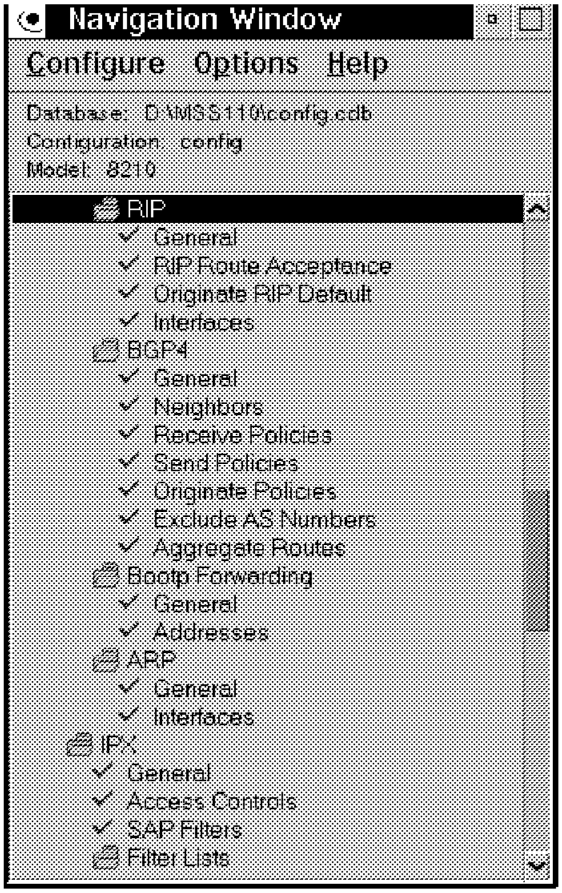
- The routes RIP always accepts
- The RIP receive policies defined per IP interface address (see 4 on page 178)

Within the RIP Route Acceptance menu, see Figure 102 on page 178, you can define which net, subnet, and host routes that RIP will accept independent of the parameters set in RIP - Parameters Per IP Address (see Figure 104 on page 179). If no Route Acceptance routing information has been entered, the receipt of RIP routes is controlled by the interface receive policies.

Note: Defining 0.0.0.0 in the Route Acceptance menu triggers RIP to always accept the default route.

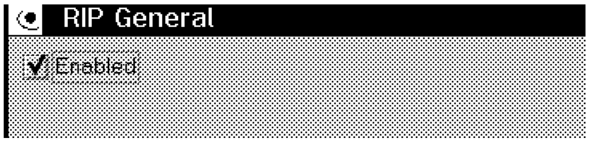
3 Control broadcast of the default route

Figure 103 on page 178 depicts the configuration options to control the broadcast of a Default Route. The configuration choices are: no broadcasting of the default route, an unconditional broadcast of the default route, or a conditional broadcast depending on routing information learned from BGP and/or OSPF.



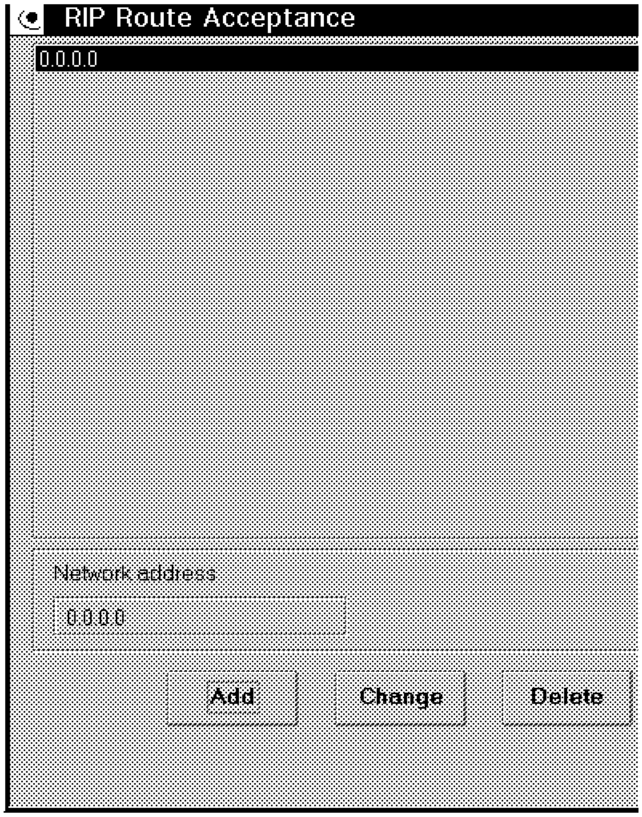
Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 100. RIP Configuration Options



☑ Enabled

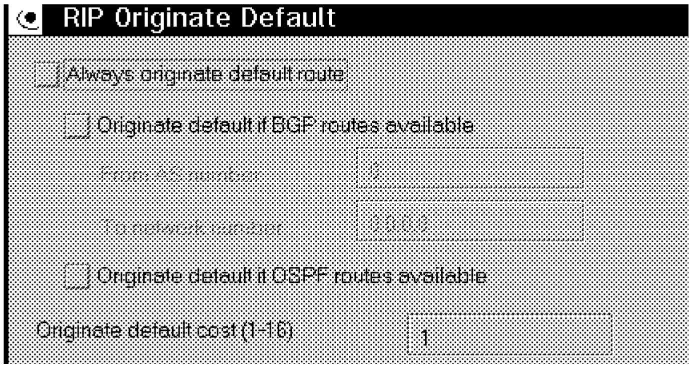
Figure 101. Enable RIP Globally



0.0.0.0

Network address: 0.0.0.0

Figure 102. Accepted Routes



□ Always originate default route

□ Originate default if BGP routes available

From AS number: 0

To network number: 0.0.0.0

□ Originate default if OSPF routes available

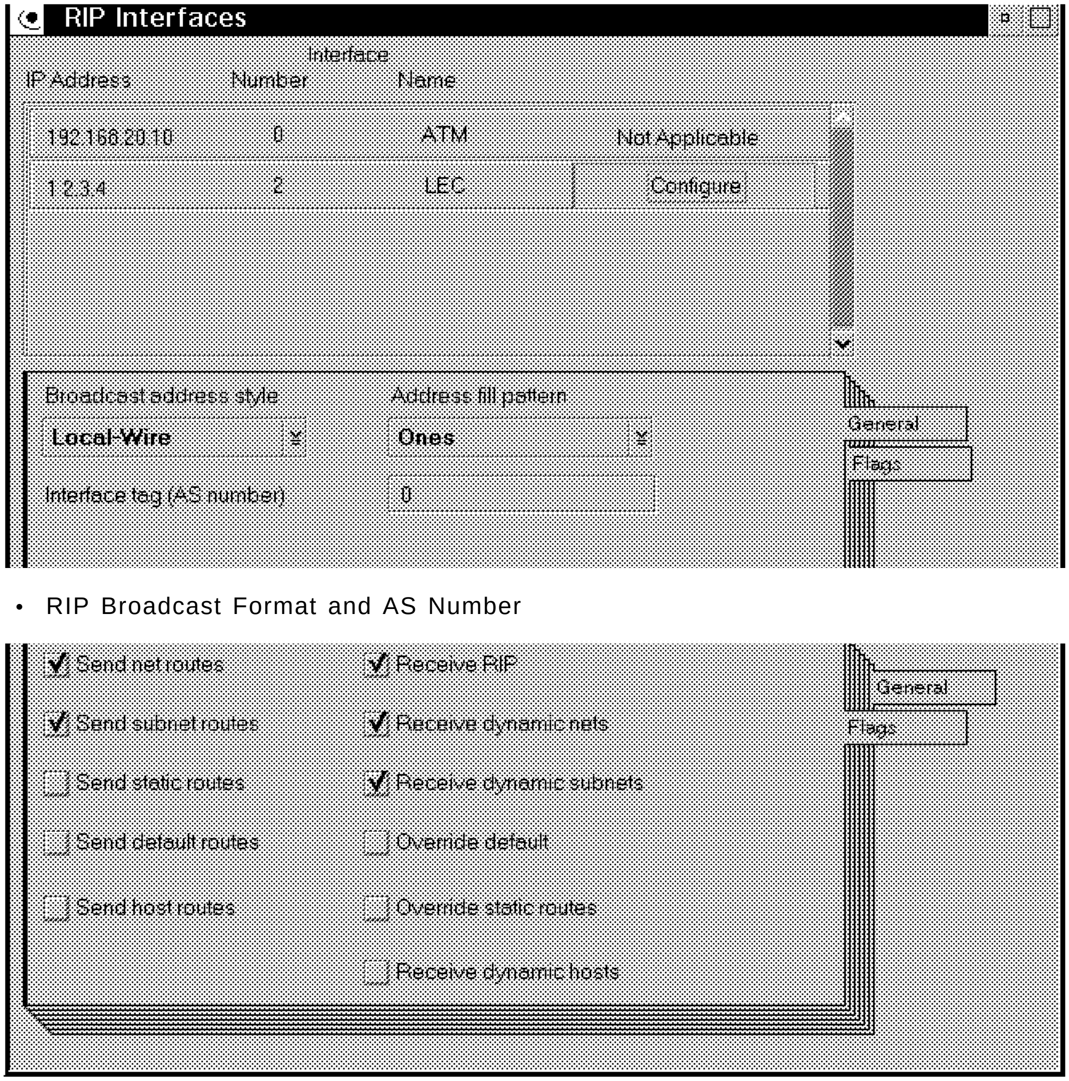
Originate default cost (1-16): 1

Figure 103. RIP Originate Default

4 Configure RIP interfaces

After enabling RIP on box-level, RIP needs to be configured [2] on each of the IP interfaces on which RIP functions are required. A number of submenus (see Figure 109 on page 186) are available for this purpose. These submenus become available after clicking on Interfaces in the Navigation Window and selecting either General or Flags for the appropriate RIP address.

[2] RIP is by default enabled on all IP interfaces.



RIP Interfaces

IP Address	Interface Number	Interface Name	
192.168.20.10	0	ATM	Not Applicable
1.2.3.4	2	LEC	Configure

Broadcast address style: Local-Wire

Address fill pattern: Ones

Interface tag (AS number): 0

- RIP Broadcast Format and AS Number

☑ Send net routes
☑ Send subnet routes
☐ Send static routes
☐ Send default routes
☐ Send host routes

☑ Receive RIP
☑ Receive dynamic nets
☑ Receive dynamic subnets
☐ Override default
☐ Override static routes
☐ Receive dynamic hosts

- RIP Routes Broadcast and Receipt

Figure 104. RIP Interface Parameters

a. Specify broadcast type and AS number

Selecting General enables you to specify the type of broadcast and the number of the Autonomous System (AS) to which the IP address being configured belongs.

Note: Unless you are using BGP to connect your routing domain to another AS, you can always set the AS field to 0.

b. Specify RIP behavior

Selecting Flags enables you to specify which types of routes (net, subnet, host, and default route) will be broadcast and/or accepted for the IP address being configured.

Steps 1 on page 177 to 3 on page 177 are global configuration steps that need to be done only once. Step 4 on page 178 needs to be repeated for every IP interface address for which you want RIP functions. The RIP interface functions are configured independently for every IP address assigned to any of the LEC interfaces

Note: Up to 32 IP addresses can be configured to a single LEC interface.

7.3 Using OSPF

OSPF is a relatively new routing protocol and, for that reason, does not have the installed base of RIP. However, most large internetworks currently in design or being implemented use OSPF as the major routing protocol. This is because OSPF overcomes many of the limitations of RIP:

- It can be used on emulated LANs and within Classical IP subnets.

> **LAN Emulation and Classical IP Support**
>
> OSPF is supported on both LAN emulation and Classical IP subnets.

- It supports variable length subnet masks.
- It supports alternate routes based on IP type of service (not supported by the IBM 8210).
- It supports equal-cost multipath routes.
- There is no hop count limit.
- Network convergence is fast and can be designed for by use of small areas.
- Only network topology changes are advertised thus reducing network traffic.

Our recommendation is to use OSPF as the major routing protocol for all new network designs and replace RIP routed networks when possible. If this cannot be done, an OSPF backbone should be used to connect the existing RIP routed networks.

When designing an OSPF network, consideration should be given to partitioning the network into backbone and non-backbone areas. Selecting optimum area sizes can significantly reduce the amount of OSPF traffic passing over individual links. This can be important if links are low-speed point-to-point.

Areas may also be defined as stub areas so that OSPF ASE link advertisements are not flooded into them.

Consideration should also be given to the IP address ranges being used within each area. The IBM 8210 allows users to define the address range being used within an area. Using consecutive subnets allows route aggregation (supernetting) to limit the number of routes advertised and reduces the size of routing tables in routers in other areas. For details, see “OSPF Advertisement of Routes” on page 507.

7.3.1 OSPF Areas

One of the key design decisions to be made for an OSPF network is how it should be partitioned into *areas.* All OSPF networks must consist of at least one area, the backbone, plus other areas that use the backbone for inter-area traffic.

Many small to medium size networks may consist of just a backbone area. In this case all routers are configured as part of area 0.0.0.0 and there will be no *area border routers*. If there are default routes or if exterior gateway protocols are being used to communicate with other ASs, then one or more routers must act as AS boundary routers.

Backbone-only networks are acceptable only up to a certain size. As area size increases, the number of link-state advertisements that must be flooded over all links increases and, particularly when point-to-point links are concerned, there is a time when the network must be partitioned.

It is difficult to define when a network reaches the size where it must be configured into multiple areas. It is dependent on overall network topology, speed of links and administrative factors.

Some possibly useful guidelines that can be used in making this decision are as follows:

- OSPF has a design point of 100 routers per area maximum.
- OSPF areas should be defined, where possible, in line with existing administrative boundaries.
- OSPF areas should be defined such that local administration errors will have a minimum impact on the overall network.
- OSPF areas should be defined to minimize traffic over point-to-point links.
- OSPF areas should, where possible, be defined to keep the backbone contiguous and hence avoid virtual links.

Once the decision is made on area partitioning, then routers can be configured with area information.

Each router must be given an *area identifier*. The backbone must use identifier 0.0.0.0, and other areas conventionally use an identifier corresponding to the network number for the networks or subnetworks included in this area.

In addition, areas must be defined as *transit* or *stub*. The normal choice for an area would be transit (the backbone *must* be transit), although stub can be chosen for areas with, for example, a single area border router exit points. The advantage of configuring an area as a stub is that a default route is used to define the exit point from the stub and no AS external link advertisements are flooded into it. This can significantly reduce OSPF link-state information within an area if large numbers of external routes are available to the OSPF network.

The final choice that must be made for an area is the *authentication scheme* to be used for all OSPF message exchanges within the area. The OSPF common header includes two fields, *authentication type* and *authentication data*. Two authentication types are possible: *none*, in which case no authentication data is passed in the data field, or *simple*, in which case an eight-byte password is passed in the data field. All routers within an area must be configured with the same authentication type. All neighbor routers must use the same password value on the network that interconnects them.

7.3.2 OSPF Interfaces

OSPF area configuration for a router requires that each interface to an area be defined by its IP address and by a number of other parameters. Care should be taken with area border routers to ensure that interfaces are assigned to the correct area.

All router interfaces to the same network must be assigned the same value for password if an authentication type of *simple* is chosen. If not, OSPF messages will be rejected. All router interfaces to the same network must be assigned the same timer values for *HelloInterval* and *RouterDeadInterval*. The HelloInterval is the length of time in seconds between Hello packets on the interface, and the RouterDeadInterval is the number of seconds before a neighbor declares the router down if no Hellos are received. These two parameters are sent in all Hello messages. If they are not configured with the same value as in neighbor routers then the Hello messages will be rejected. It is essential, therefore, that all routers on a network use the same values. It is recommended, for this reason, that the defaults be taken wherever possible.

Each router interface must be assigned a *cost* based on the type of network link. This cost is used in the determination of the total path cost to a destination. It is recommended to define a lowest cost on your high-speed ATM connections.

If the interface is to a non-broadcast network, it is necessary to configure the IP addresses of neighbor routers on the network. This is to allow the router to send OSPF messages to neighbors using specific IP addressing.

7.3.3 Area Ranges

Area ranges need only be configured on area border routers. They represent the address ranges that are contained within the area and which will be summarized in summary links advertisements to other areas attached to the area border router.

Configuration of area ranges is optional. When using (consecutive) address ranges that can be aggregated in a single route, or a limited number of routes, defining area ranges helps to minimize the number of routes advertised, thereby reducing the routing table within routers in other areas.

7.3.4 Virtual Links

Virtual links are necessary to connect non-contiguous sections of an OSPF backbone. Virtual links are configured between two area border routers through a non-backbone area by specifying the IP address of the destination area border router (the virtual neighbor identifier) and the area through which the virtual link will pass. They are configured for the backbone area 0.0.0.0 only.

Note: Careful design of network topology should be used to avoid the use of virtual links where possible.

7.3.5 OSPF Configuration on the IBM 8210

Configuring the 8210 OSPF functions requires the configuration of box and interface specific parameters (see Figure 105 on page 183).

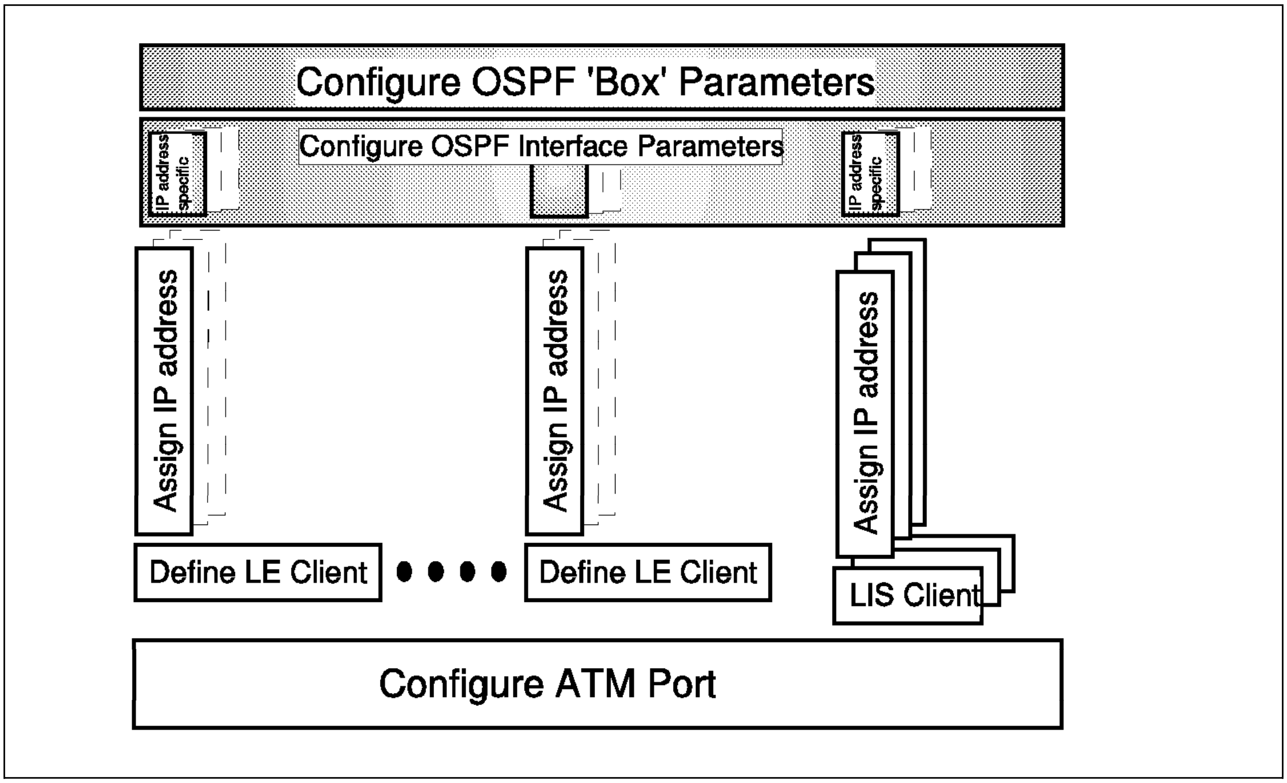
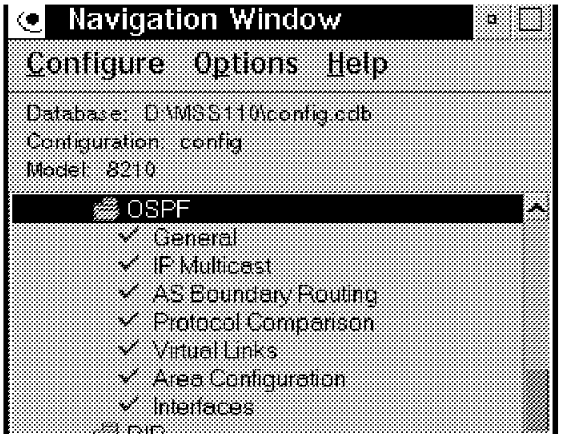


Figure 105. Configuring OSPF Functions



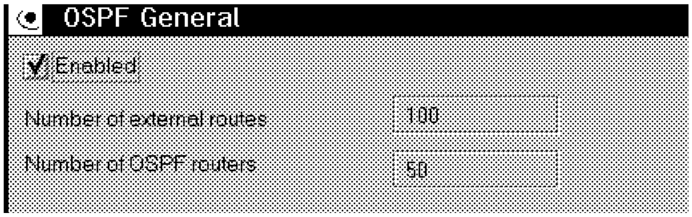
Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 106. OSPF Navigation

Assuming the appropriate IP addresses have been defined, the Configuration Program requires the following steps (see Figure 106) to complete the OSPF configuration:

1 Enable OSPF on box level

OSPF is, by default, disabled and needs to be enabled before it can be used. This configuration option, see Figure 107 on page 184, becomes available after clicking on OSPF General in the Navigation Window.


OSPF General

☑ Enabled

Number of external routes 100

Number of OSPF routers 50

Figure 107. Enable OSPF

2 Configure OSPF areas

Figure 108 on page 185 enables you to define the OSPF areas to which the 8210 connects. The two submenus become available after selecting General or Area Ranges, respectively. Parameters that need to be entered are:

a. Configure the area number, authentication type, and type of area

   All OSPF areas to which the IBM 8210 connects need to be explicitly defined. The area number must be unique and consistent with the area numbers defined on other routers within the area. The area number for the OSPF backbone is 0.0.0.0. All routers within an area must define the same values for Stub Area and Authentication.

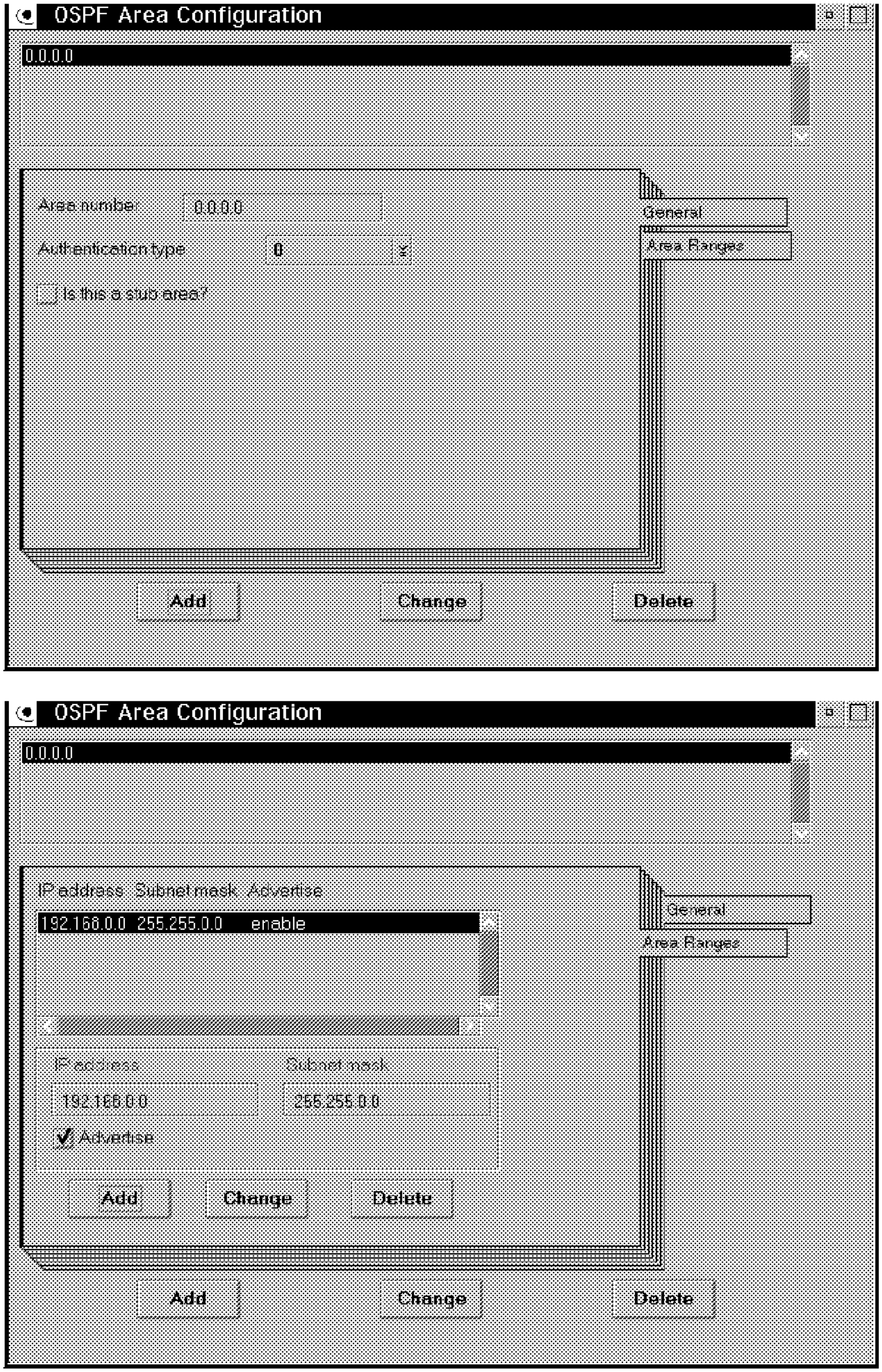
   Note: On stub area border routers, the import of Import (route) summaries can be disabled, if proper default routes have been defined (see Figure 109 on page 186).

b. Submenu B (Area Ranges), to configure the address ranges that are advertised

   OSPF area border routers may define address ranges. The address ranges entered may be a list of subnet address ranges within the same area or, for example, a single route in which all subnet address ranges have been aggregated. The latter results in less routes advertisement and reduces the routing tables in routers within other areas. Hybrid solutions, specifying both single subnet and aggregated routes, are possible as well.

3 Specify OSPF behavior on Autonomous System (AS) boundary

Figure 109 on page 186 appears after selecting AS Boundary Routing in the Navigation Window. The configuration menu enables you to control the types of routes that are imported from an external AS into the OSPF routing domain.



OSPF Area Configuration

0.0.0.0

Area number: 0.0.0.0

Authentication type: 0

□ Is this a stub area?

OSPF Area Configuration

0.0.0.0

IP address Subnet mask Advertise

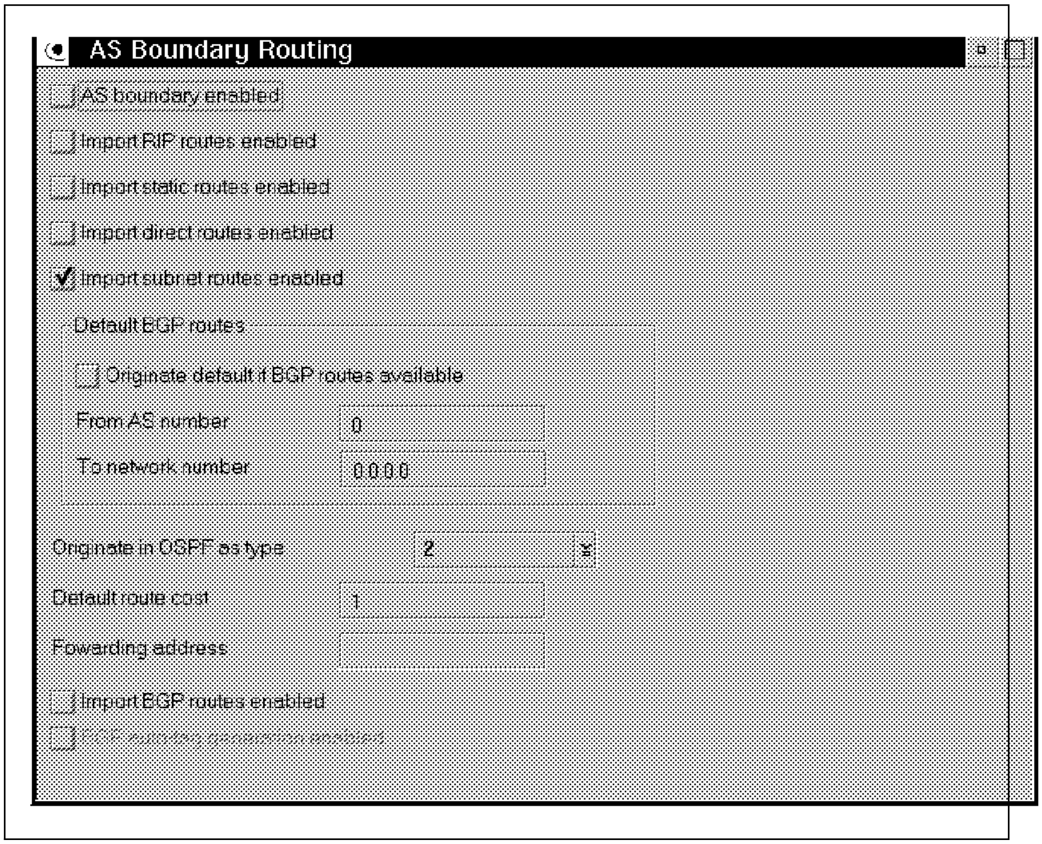
192.168.0.0 255.255.0.0 enable

IP address: 192.168.0.0

Subnet mask: 255.255.0.0

☑ Advertise

Figure 108. OSPF Areas



AS Boundary Routing

□ AS boundary enabled

□ Import RIP routes enabled

□ Import static routes enabled

□ Import direct routes enabled

☑ Import subnet routes enabled

Default BGP routes

□ Originate default if BGP routes available

From AS number: 0

To network number: 0.0.0.0

Originate in OSPF as type: 2

Default route cost: 1

Fowarding address:

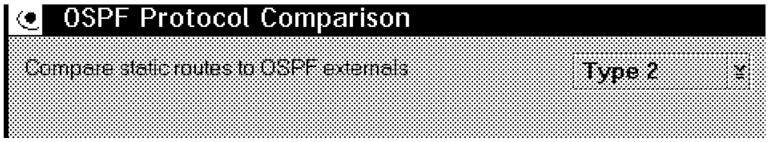
□ Import BGP routes enabled

□ [illegible] generation enabled

Figure 109. AS Boundary Routing

4 Compare static routes to OSPF external

Figure 110 appears after selecting Protocol Comparison in the Navigation Window. The configuration menu enables you to specify whether static routes are treated as Link State Advertisements (LSAs) type 1 or type 2. For details about LSA type 1 and type 2, see B.5.4.7, "OSPF Routing Policies" on page 507.



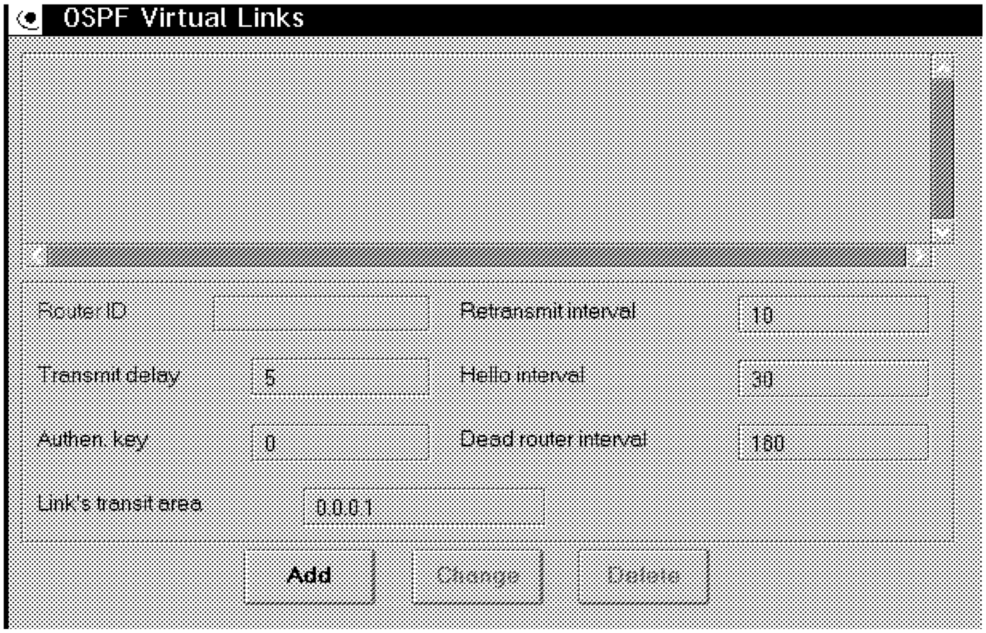
OSPF Protocol Comparison

Compare static routes to OSPF externals: Type 2

Figure 110. Protocol Comparison

5 Define virtual links

In the situation that your backbone area is non-contiguous, virtual links have to be defined. Figure 111 on page 187 appears after selecting Virtual Links in the Navigation Window. Make sure that the parameters are set to the same values on both ends of the virtual links. For details on OSPF virtual links, see 7.3.4, "Virtual Links" on page 182.



Router ID

Retransmit interval 10

Transmit delay 5

Hello interval 30

Authen. key 0

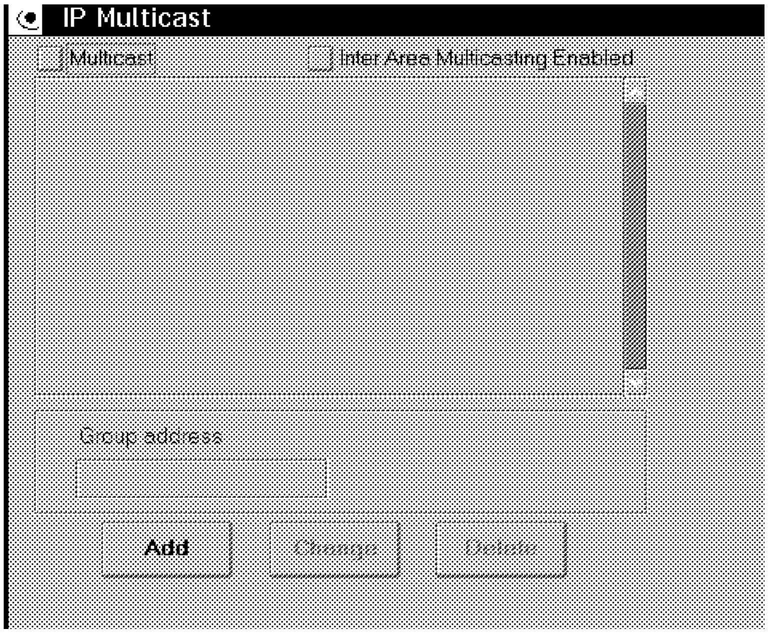
Dead router interval 180

Link's transit area 0.0.0.1

Figure 111. OSPF Virtual Links

6 Configure OSPF multicast forwarding

Figure 112 shows the configuration options to enable the MSS Server to participate in the forwarding of multicast frames. Configuration options exist to participate in multicasting, perform multicasting confined within an OSPF area, and/or perform inter-area multicasting.



□ Multicast □ Inter Area Multicasting Enabled

Group address

Figure 112. OSPF Multicast Forwarding and Group Address

For details on the OSPF multicast facility see B.5.3, "Multicast Extensions to OSPF (MOSPF)" on page 503.

7 Configure OSPF group addresses

In addition to defining multicast forwarding Figure 112 on page 187 also shows how to define an OSPF multicast group address. When defined the MSS Server itself joins the group and responds to queries (such as PING and SNMP) addressed to the group′s address. It is not necessary to define a group on the MSS Server to forward multicast packets or poll attached networks. Make sure the group address is a valid Class-D address.

For details on the OSPF multicast facility see B.5.3, "Multicast Extensions to OSPF (MOSPF)" on page 503.

8 Configure OSPF interfaces

After enabling OSPF and box-level (see Figure 107 on page 184) and defining each of the areas to which the 8210 attaches, OSPF needs to be enabled on each of the IP interfaces on which OSPF functions are required. A number of submenus (see Figure 113 on page 189) are available for this purpose.

a. Enable OSPF and define area-specific information

   OSPF needs to be explicitly enabled for each ELAN and/or LIS on which OSPF dynamics are required. When enabling an interface one has to indicate the OSPF area number used by the respective LE or LIS client.

   The Cost value enables you to prioritize interfaces within an area.

   The Priority field plays a role during the election of the designated router. A non-zero Priority makes the router eligible for designated router on this interface. Routers with higher priority are favored to be designated router. Other routers must be configured with zero router priority.

   If authentication has been enabled for this area (see Figure 108 on page 185), make sure that neighbor routers use the same authentication key. For details on the interface parameters, see 7.3.2, "OSPF Interfaces" on page 182. To simplify the configuration process, use defaults when possible.

b. Set OSPF multicast-specific information

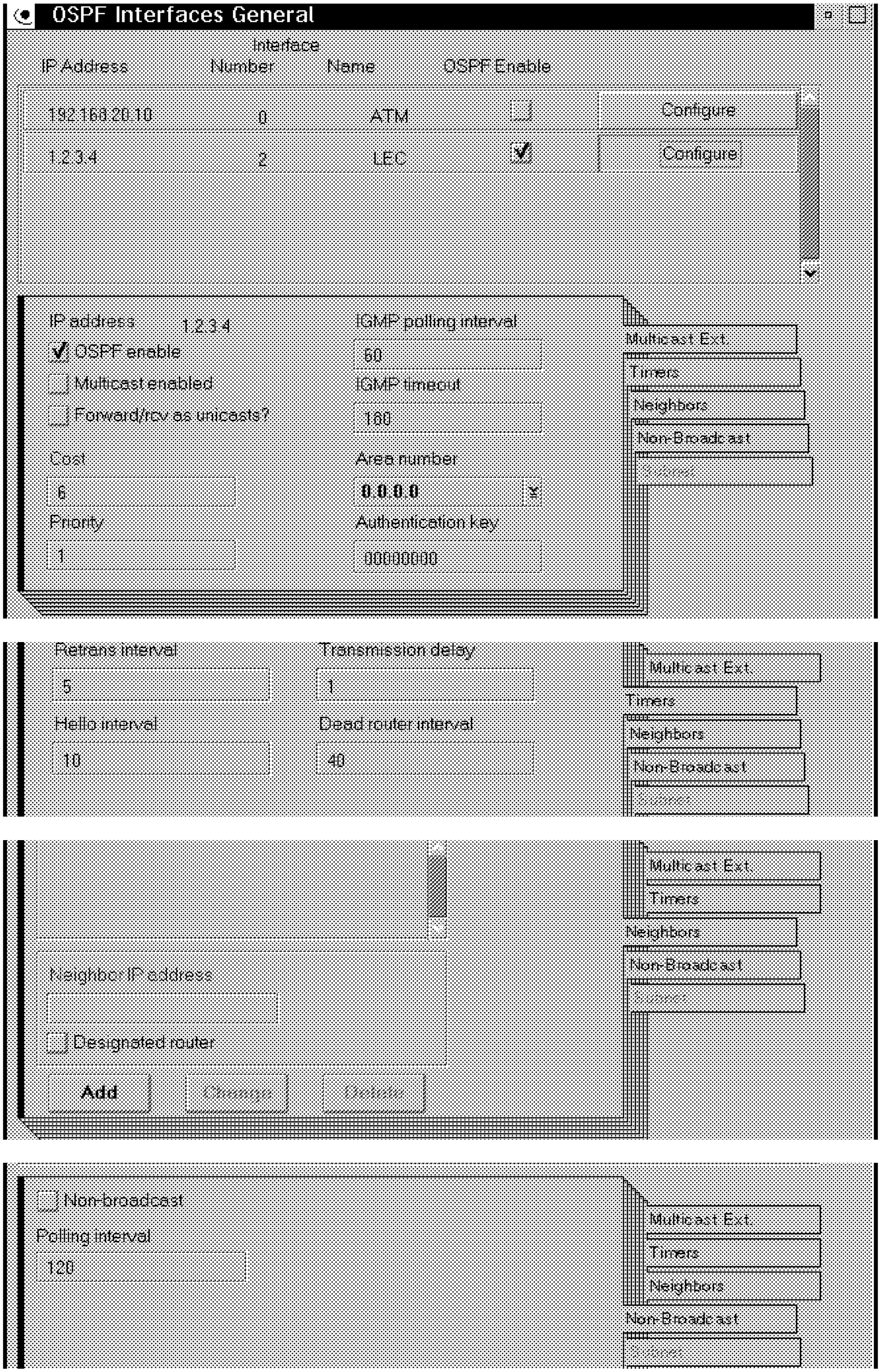
   OSPF interface multicast facilities need to be defined in addition to enabling global multicast support. To control the transmission of OSPF multicast traffic, IGMP (Internet Group Management Protocol) is used. Timers exist to control the polling of IGMP stations. For details on the OSPF multicast facilities, see B.5.3, "Multicast Extensions to OSPF (MOSPF)" on page 503.

c. Set timers

   Neighbor OSPF routers exchange control messages periodically. Hello and Dead router intervals must be the same on all OSPF routers within a given area.

d. Define neighbors (Classical IP only)

   The option exists to define neighbor OSPF routers reachable via each of the OSPF IP interfaces. This is to allow the router to send OSPF messages to neighbors using specific IP addressing (rather than



OSPF Interfaces General

IP Address	Interface Number	Name	OSPF Enable	
192.168.20.10	0	ATM	☐	Configure
1.2.3.4	2	LEC	☑	Configure

IP address 1.2.3.4

☑ OSPF enable

☐ Multicast enabled

☐ Forward/rcv as unicasts?

Cost 6

Priority 1

IGMP polling interval 60

IGMP timeout 180

Area number 0.0.0.0

Authentication key 00000000

Retrans interval 5

Transmission delay 1

Hello interval 10

Dead router interval 40

Neighbor IP address

☐ Designated router

☐ Non-broadcast

Polling interval 120

Figure 113. OSPF Interface Parameters

broadcasting Class-D group addresses). Hereby one can specify which of the neighbor routers is eligible to become designated router (DR).

Note: The definition of OSPF neighbors is required on Classical IP interfaces only. See the discussion in 8e.

e. Specify type of network interface (Classical IP)

When configuring a Classical IP interface for OSPF one has the option to define the type of subnet model which is used. Hereby two options exist: the non-broadcast multi-access (NBMA) or the point-to-multipoint (PtMP) model.

PtMP is the default network model supporting any connectivity pattern. NBMA requires a fully meshed (VCCs between each pair of routers) configuration. Because NBMA is more efficient, its use is recommended, especially in large network constructs. Note that Classical IP requires direct connections between adjacent nodes anyway.

When using PtMP, make sure that at least one side of each PtP connection defines its neighbor.

When using NBMA, make sure that:

- All routers define the network as NBMA
- All routers, except DR-eligible routers, define a zero priority
- DR-eligible routers define all neighbor routers, indicating which neighbors are (also) DR-eligble

Steps 1 on page 183 to 2 on page 184 are global configuration steps that need to be done only once. Step 8 on page 188 needs to be repeated for every IP interface address for which you want OSPF functions. The OSPF interface functions are configured independently for every IP address assigned to any of the LEC or Classical IP interfaces.

Note: Up to 32 IP addresses can be configured to a single logical interface.

7.4 Using BGP Version 4

BGP Version 4 is the newest and the preferred exterior gateway protocol. It was developed on the basis of experience gained with EGP in the Internet. It, therefore, offers several enhancements to the features offered by EGP. BGP can be used on emulated LANs and within Classical IP subnets.

LAN Emulation and Classical IP Support

BGP is supported on both LAN emulation and Classical IP subnets.

In particular it provides:

- Advertisement of all routes used by a BGP router, not only those within the local AS
- Full paths to destinations which can be through multiple ASs
- Information about how a route was learned by a BGP router
- Next hop information which need not be the neighbor router on the same network

- Intra-area metrics to allow selection to be made between alternate routes to a destination
- Support for neighbor routers that are not on the same common network
- Support for *classless* routes which allows route aggregation leading to smaller routing tables

Provision of this additional information by an originating router allows complex policy decisions to be made based on the information.

BGP-4 should be considered when you wish to use public IP services to communicate with other organizations whose networks reside in separate ASs. This requires that you have registered your organization, have been issued an AS number, and have registered the network numbers used within your organization

Using BGP requires that routers on both ends of a BGP connection support the same BPG version (that is, Version 4).

7.4.1 AS Numbers

BGP can only be configured when the local AS has been allocated an *AS number*, and the identity of the ASs to which neighbor BGP routers belong is known. If this information is not available then it is not possible to proceed. ASs are designated numerically in the range 1 to 65534. AS numbers for an organization must be registered publicly before they can be used.

AS numbers are used to qualify the local router originating BGP packets and also to fully define a neighbor router whose routes are being advertised. AS numbers are further used as a filter for routes exported between BGP and interior gateway protocols.

7.4.2 BGP Configuration on the IBM 8210

Configuring the 8210 BGP functions requires the configuration of box-level parameters only (see Figure 114 on page 192), and no interface specific parameters are required.

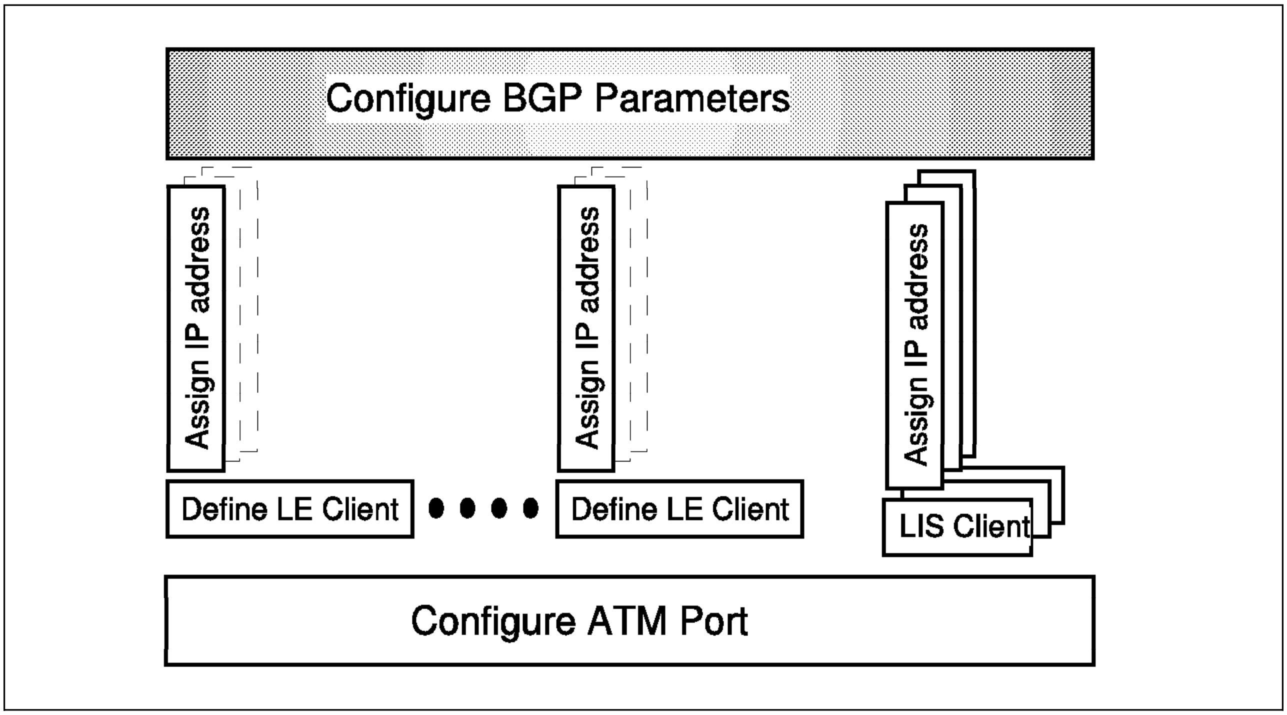
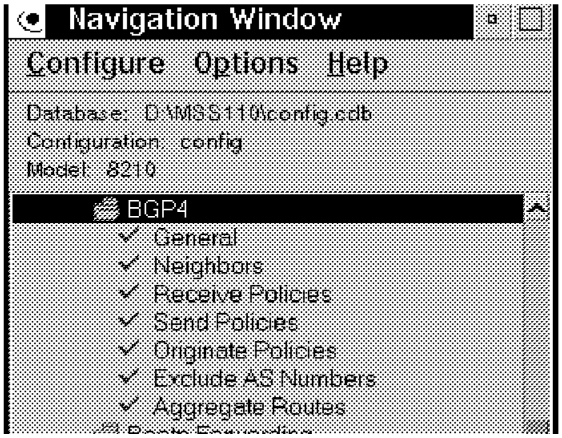


Figure 114. Configuring BGP Functions

Assuming the appropriate IP addresses have been defined, the Configuration Program requires the following steps (see Figure 115) to complete the BGP configuration:

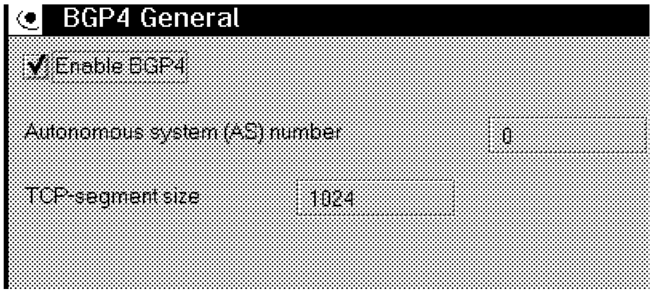
1 Enable BGP on box level

BGP is, by default, disabled and needs to be enabled before it can be used. This configuration option, see Figure 116, becomes available after clicking on BGP General in the Navigation Window. In the same figure define the local AS number and the TCP segment size for communication to BGP neighbors.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 115. Navigation Window



☑ Enable BGP4

Autonomous system (AS) number: 0

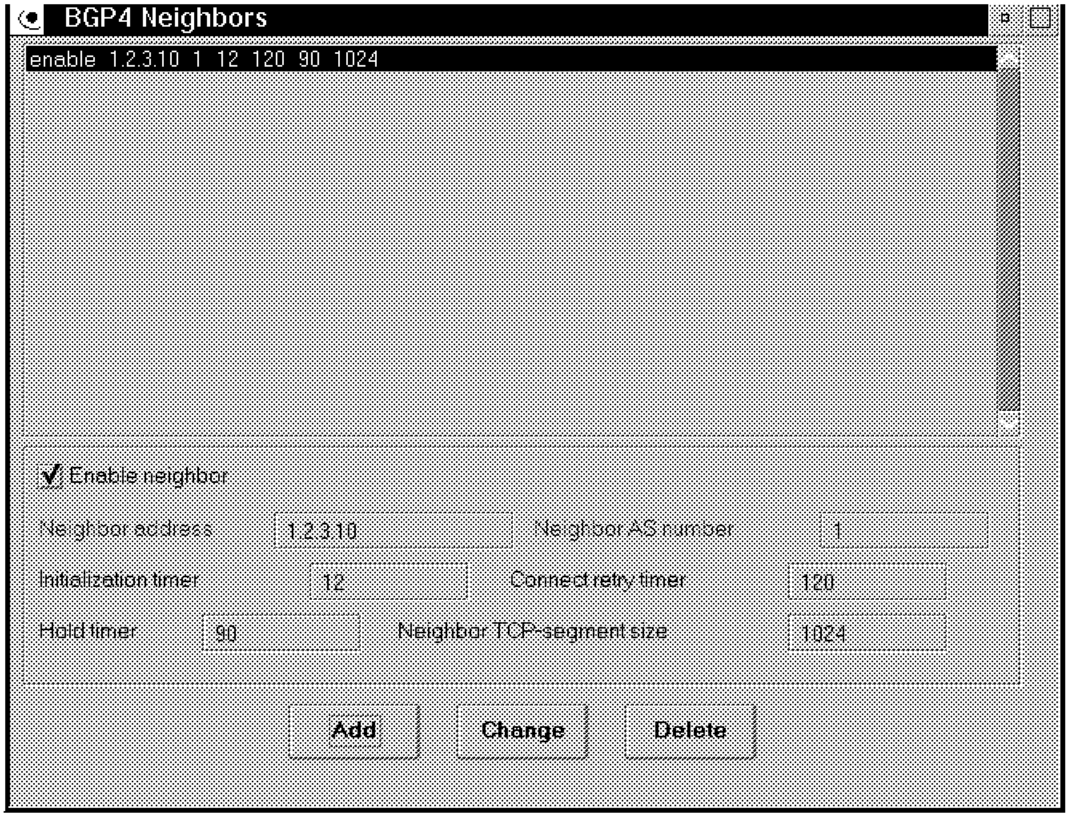
TCP-segment size: 1024

Figure 116. Enable BGP

2 Define BGP neighbors

After BGP has been enabled, *BGP neighbors* must be defined (see Figure 117 on page 193). BGP neighbors are defined by their IP address. BGP neighbors can be of two types: *enabled* and *disabled*. Enabled neighbors are fully defined and communication with them is initiated on

BGP startup. Communications with disabled neighbors can be initiated by the neighbor only. AS-external neighbors must be adjacent (that is, attach to the same (sub)net, and AS-internal neighbors do not have to be adjacent.



BGP4 Neighbors

enable 1.2.3.10 1 12 120 90 1024

☑ Enable neighbor

Neighbor address 1.2.3.10 Neighbor AS number 1

Initialization timer 12 Connect retry timer 120

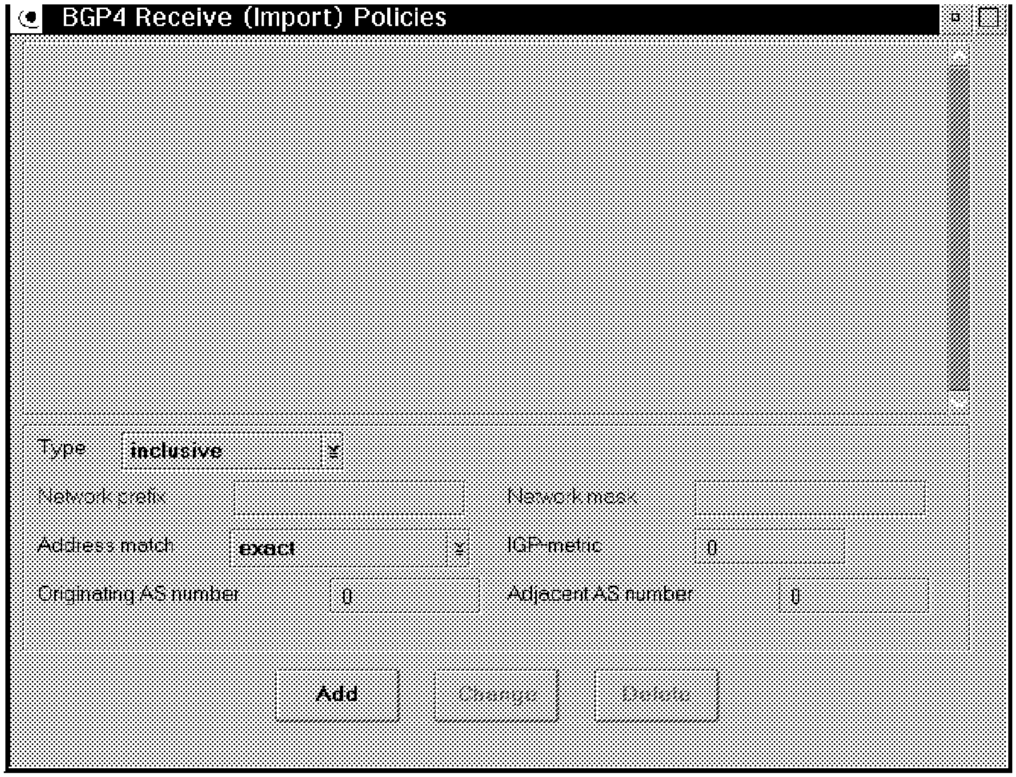
Hold timer 90 Neighbor TCP-segment size 1024

Figure 117. BGP Neighbors

3 Define receive policies

Figure 118 on page 194 shows how the BGP receive policies can be defined. The receive policies control which BGP-received routes will be injected into the router routing table and forwarded into the RIP and/or OSPF domain to which the IBM 8210 router connects. Inclusive routes are accepted, while exclusive routes will not be accepted. The originating AS number indicates the AS that has originated the routes, while the adjacent AS number indicates from which AS the IBM 8210 has received the route.

BGP can be configured to permit the import of the default route (0.0.0.0) from BGP neighbors. Before enabling this option, make sure you understand the operational consequences.



Type: inclusive

Network prefix:

Network mask:

Address match: exact

IGP-metric: 0

Originating AS number: 0

Adjacent AS number: 0

Figure 118. BGP - Receive Policies

4 Define send policies

Figure 119 on page 195 shows how the BGP send policies can be defined. The send policies control which BGP-received routes are forwarded to BGP neighbors.

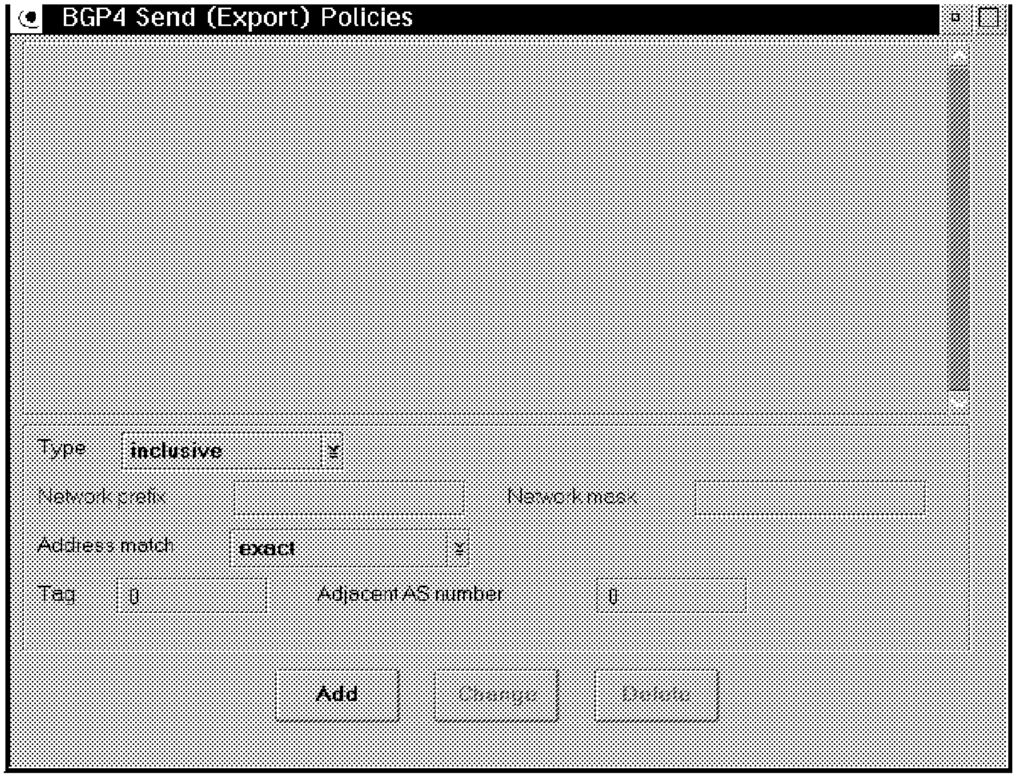
5 Define originate policies

Figure 120 on page 195 shows how the BGP originate policies can be defined. The originate policies control which RIP and/or OSPF learned routes are forwarded to BGP neighbors.

BGP neighbors can be configured to permit the export of the default route (0.0.0.0) to BGP neighbors. It is not normal to enable this option. It should only be used when the operational consequences are well understood.

6 Exclude AS numbers

Figure 121 on page 196 enables you to define remote AS numbers from which no routes will be included into the local routing table.



Type: inclusive

Network prefix

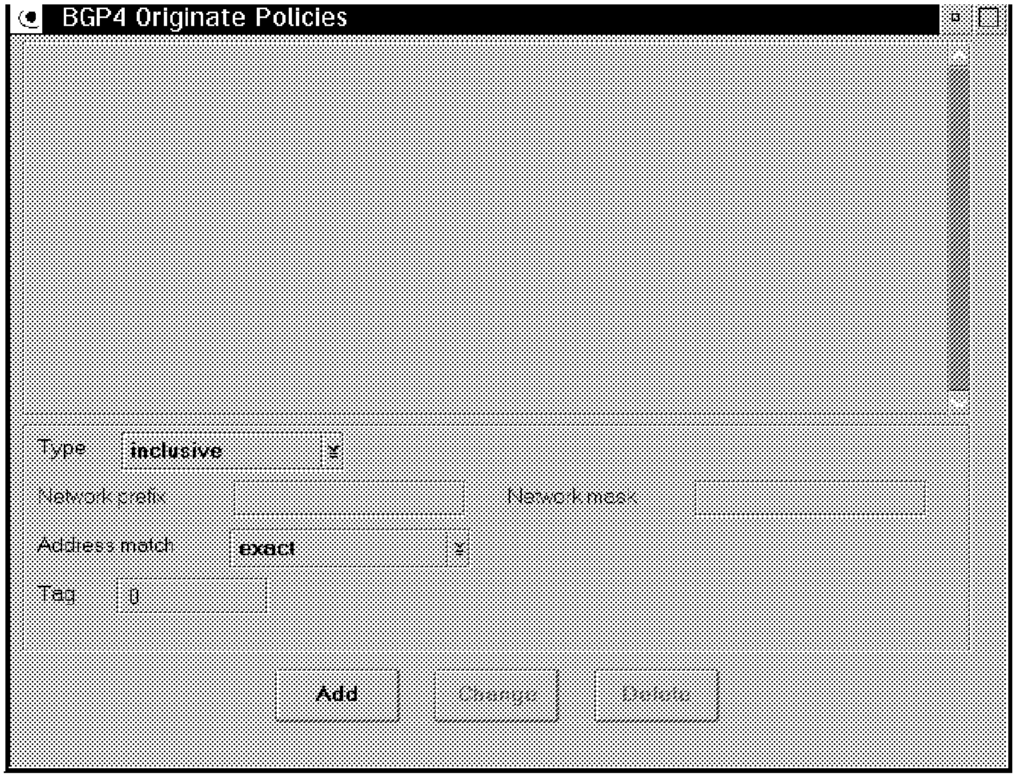
Network mask

Address match: exact

Tag: 0

Adjacent AS number: 0

Figure 119. BGP - Send Policies



Type: inclusive

Network prefix

Network mask

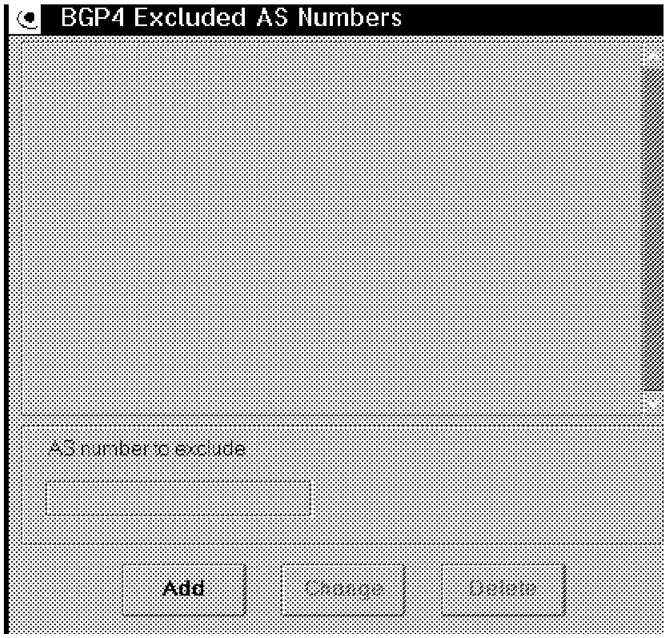
Address match: exact

Tag: 0

Figure 120. BGP - Originate Policies

7 Configure route aggregation

An important feature of BGP-4 is route aggregation. Route aggregation minimizes the number of routes advertised and, hereby, reduces the size of routing tables required in other ASs. Figure 122 on page 196 shows how BGP aggregated routes can be defined.



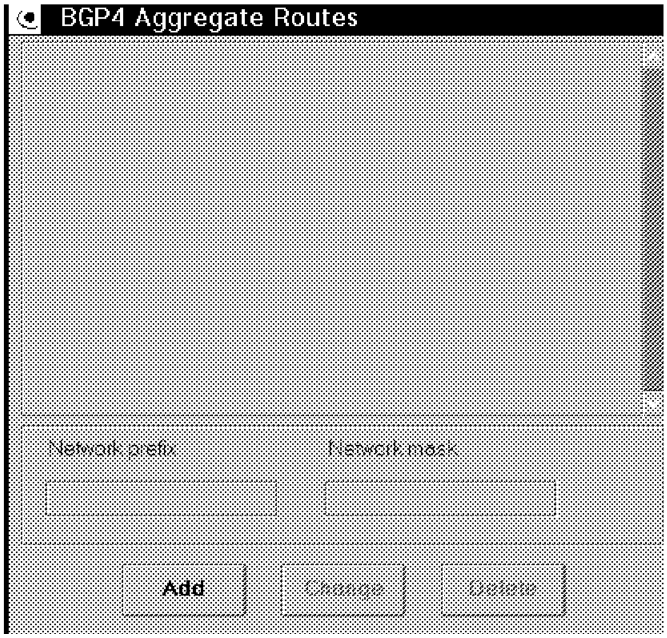


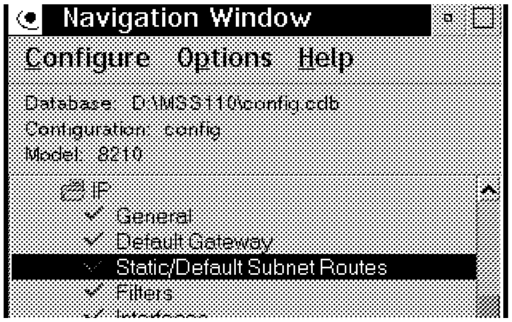
Figure 121. BGP - Exclude AS Numbers

Figure 122. BGP - Aggregate Routes

Steps 1 on page 192 to 7 on page 195 are global configuration steps that need to be done only once. No IP address specific configuration is required.

7.5 Static Routes

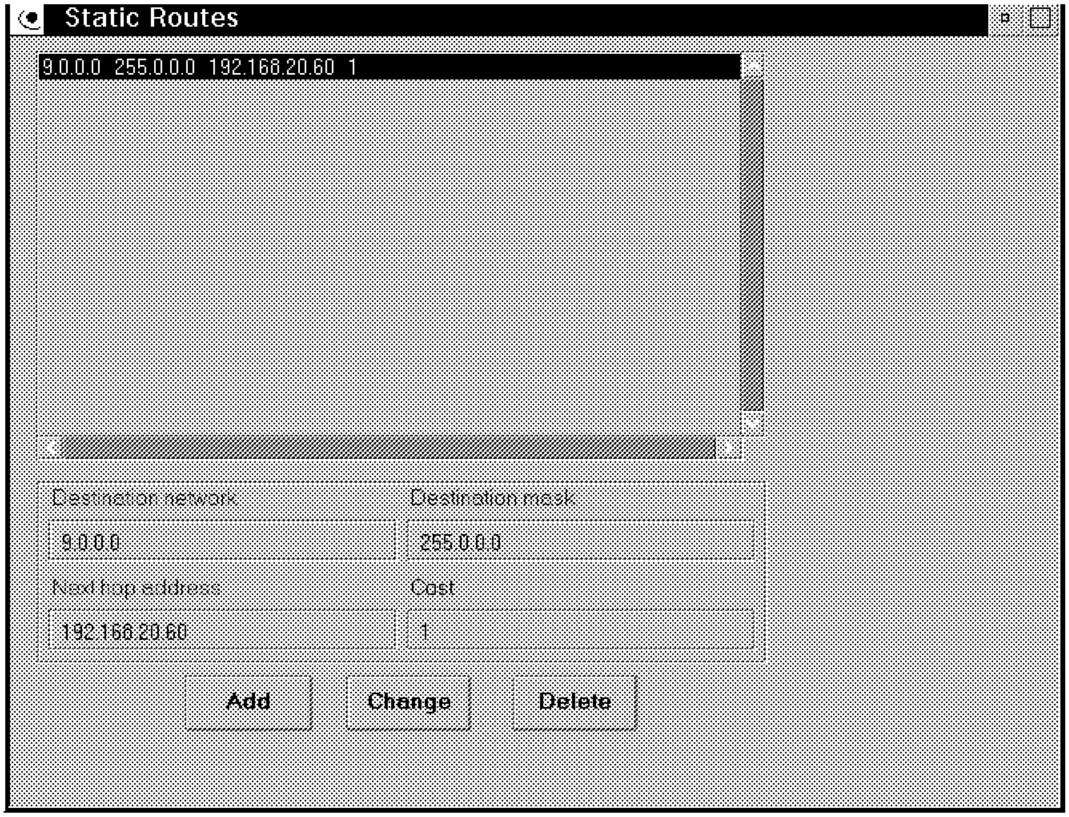
Defining static routes on IP routers is not recommended because of the administrative overhead, the chances of making errors, and the inflexibility of such a mechanism.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 123. Navigation Window

However, static routes are essential when defining a default route (for example, to minimize the number of entries in your routing tables) to define routes to networks accessible via passive routers (that is, not running a routing protocol) and to define routes to networks or ASs where a routing protocol is undesirable for reasons of link cost. The IBM 8210 router allows you to define static routes.



9.0.0.0 255.0.0.0 192.168.20.60 1

Destination network: 9.0.0.0

Destination mask: 255.0.0.0

Next hop address: 192.168.20.60

Cost: 1

Figure 124. IP Static Routes

Figure 124 on page 197 depicts how to configure static routes. This menu becomes available after clicking on Static/Default Subnet Routes in the Navigation Window (Figure 123).

Default, host, network, and/or subnet routes can be defined. A default route is indicated by a network address and a subnet mask of all zeroes (0.0.0.0). A host route is indicated by a non-zero network address and a subnet mask of 255.255.255.255. A network route is indicated by entering the network number appended by zeroes (for example, 9.0.0.0) and a subnet mask reflecting the network class (for example, 255.0.0.0 for Class A networks). A subnet route is indicated by entering the subnet number appended by zeros (for example, 9.132.36.0), with the subnet mask reflecting the subnetting chosen (for example, 255.255.255.0 for Class C subnetting).

Note: When RIP is employed as the interior gateway protocol, make sure that a single subnet mask is used within your network.

7.6 Routing Protocols Interoperability

When combinations of routing protocols are used on the IBM 8210, it is necessary to specify how to pass routes learned by one protocol to another.

Examples of situations where this is required include:

- Advertising static/default route
- Passing route information between RIP and OSPF gateway protocols

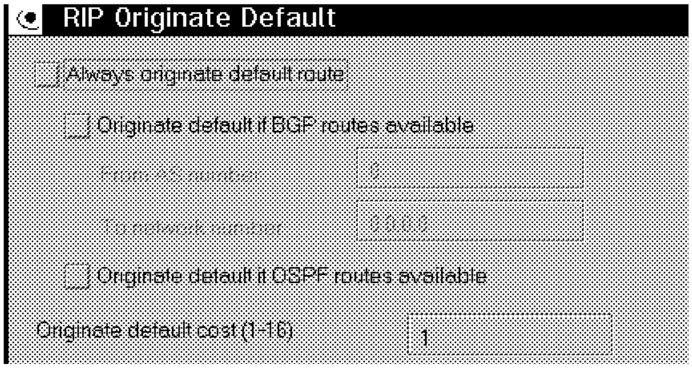
- Passing route information between the interior gateway protocol(s) (RIP/OSPF) and BGP

The principle behind route export is simple but there are a number of implementation details that must be understood for the IBM 8210.

7.6.1 RIP Specifics

The following rules apply when there is both a RIP route and a route from another protocol to the same destination:

- The RIP route will override a BGP route.
- The RIP route will be overridden by an OSPF internal route.
- The RIP route will override a static route if the RIP route′s metric is less.
- The RIP route will override a default route if the RIP route′s metric is less.
- The RIP route will override an existing OSPF Autonomous System external (ASE) type 1 route if the OSPF external comparison switch is set to type 1, and the RIP metric is lower than the OSPF metric.
- The RIP route will override an existing OSPF Autonomous System external (ASE) type 2 route if the OSPF external comparison switch is set to type 1, or the OSPF external comparison switch is set to type 2, and the RIP metric is lower than the OSPF metric.



RIP Originate Default

□ Always originate default route

□ Originate default if BGP routes available

From AS number: 0

To network number: 0.0.0.0

□ Originate default if OSPF routes available

Originate default cost (1-16): 1

Figure 125. RIP - General Parameters

The IBM 8210 router may advertise itself as the default router within the RIP routing domain depending on the values configured within the Originate Default Route field (see Figure 125). Hereby you have the option:

- To always originate the default route.
- To originate the default route when a BGP route for a specific destination has been received. The default route will be generated if the route corresponds with a specific AS. AS 0 indicates that the AS number should not be checked.
- To originate the default route when OSPF routes are present in the routing table.

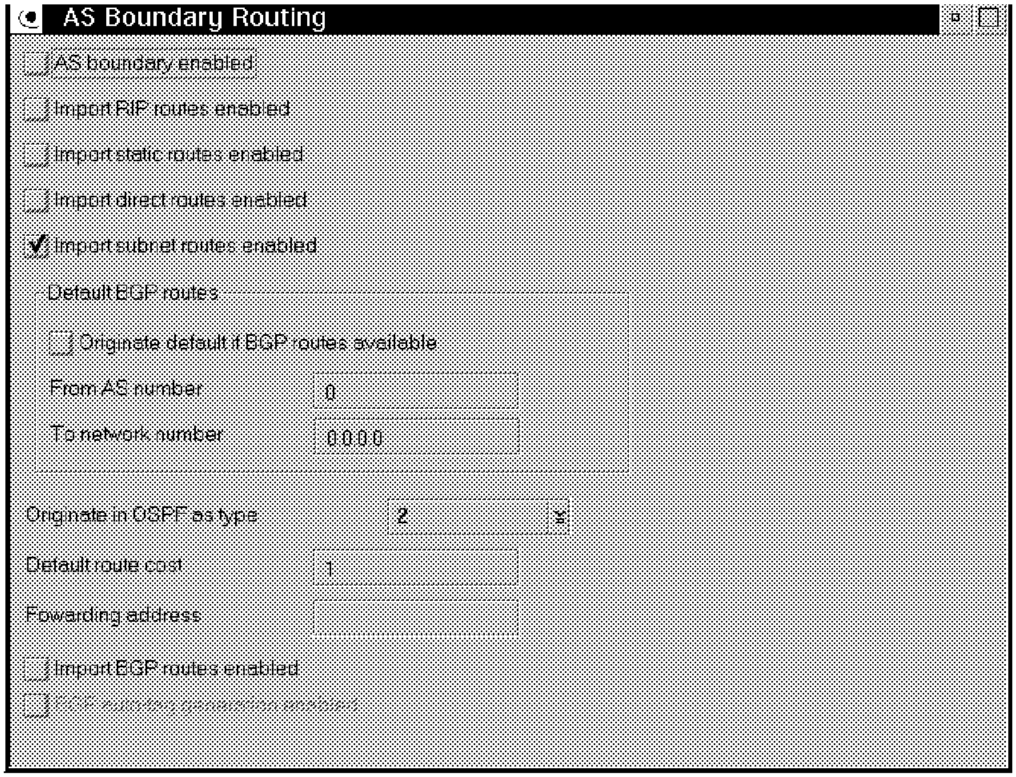
7.6.2 OSPF Specifics

The IBM 8210 has a configuration option to enable route imports. The type of routes that are imported can be specified. Options exist to import static routes, direct routes, subnet routes, RIP routes, and BGP routes.

All routes are imported as OSPF AS external routes and advertised into the OSPF network using AS external link-state advertisements.

When OSPF imports routes, it is necessary to specify additional parameters to define how the external route will be advertised into the OSPF network. These parameters are the *metric type* (1 or 2) and the *external route tag*. The metric type, if set to *2*, defines the metric to the destination to be larger than any internal path. If set to *1* it means that the metric is comparable to internal paths. It is recommended that type 2 be selected to ensure that external routes have a larger metric than internal ones. Type 1 should be selected only if the operational effect of using it is fully understood.

The external route tag provides additional information in the AS external link-state advertisement, although it is currently only defined for use with BGP. The tag, therefore, can normally be set to any value.



AS Boundary Routing

□ AS boundary enabled

□ Import RIP routes enabled

□ Import static routes enabled

□ Import direct routes enabled

☑ Import subnet routes enabled

Default BGP routes

□ Originate default if BGP routes available

From AS number: 0

To network number: 0.0.0.0

Originate in OSPF as type: 2

Default route cost: 1

Fowarding address:

□ Import BGP routes enabled

□ [illegible] generation enabled

Figure 126. OSPF - Imported/Default Routes

The IBM 8210 router may advertise itself as the default router within the OSPF routing domain depending on the values configured within the Originate Default Route field (see Figure 126). Hereby you have the option:

- To always originate the default route.
- To originate the default route when a BGP route for a specific destination has been received. The default route will be generated if the route

corresponds with a specific AS. AS 0 indicates that the AS number should not be checked.

The forwarding address indicates to which router default packets are routed. It must be a router interface address that is adjacent (that is, part of the same -sub-net) to the IBM 8210.

IBM 8210 routing policies, when multiple routes to the same destination are available, are used in the following order:

- Interior gateway protocols (RIP, OSPF) routes are preferred over BGP routes.
- OSPF routes are always preferred over other types of routes (including the route to a directly attached network if the network interface has not been enabled for OSPF).
  - OSPF intra-area routes are preferred over inter-area routes.
- RIP routes and static routes can override OSPF ASE routes.

  It can be configured if external routes will be compared to OSPF ASEs as type 1 or type 2 routes. For details, see “Importing Routes into OSPF” on page 507.

  Given the preceding definitions of the configurable OSPF ASE route, static and RIP routes may override OSPF ASE routes in the following situations:

  - The OSPF comparison is configured as type 1, and the OSPF ASE route is an OSPF ASE type 2 route.
  - The OSPF comparison is configured as type 1, and the OSPF ASE route is an OSPF ASE type 1 route but has a higher cost than the RIP or static route.
  - The OSPF comparison is configured as type 2, and the OSPF ASE route is an OSPF ASE type 2 route but has a higher cost than the RIP or static route.

7.6.3 BGP Specifics

The routing information that is sent by BGP to its BGP neighbor depends on the routing information within the IP routing table (that is, locally defined static routes and routes learned from RIP and/or OSPF) and routes received from BGP neighbors. This process is controlled by defining originate and send policies and by defining aggregated routes. For details, see 7.4.2, “BGP Configuration on the IBM 8210” on page 191.

7.6.4 Static/Default Routes

The default referred to in this subsection is the default route that is generated by BGP if the generate default route option is enabled.

Export default cannot be used to export an intra-AS default route without BGP being configured. If this is a requirement it must be met by the use of the static route (0.0.0.0) exported to the interior gateway protocol within the AS.

The default should be exported to the interior gateway protocol being used within the AS. It will become advertised when and only when BGP communications are established with a neighbor that is configured to generate a default route.

7.7 IBM 8210 Redundancy Features

One of the major benefits of using dynamic routing protocols is that when routers or routes fail in your network, the network automatically recovers, and alternate routes are found. However, there are situations in which this approach cannot be used.

In many networks, dynamic routing protocols are activated on the IP routers, while the host station uses a static route to a default gateway. The advantage of this approach is that the configuration of the end stations is simple, and their routing tables are kept to a minimum. The disadvantage of this approach is that when the default router fails, the endstations loose connectivity.

The next section discusses the automatic default IP gateway facilities for Classical IP and LAN emulation provided by the IBM 8210 Nways MSS Server.

7.7.1 Default Gateway Redundancy - Classical IP

Figure 127 depicts two 8210s on which a LIS client has been defined with the same ATM address and the same IP address (192.168.20.10) on both systems. Either 8210 can perform default IP routing for remote clients attached to the same LIS.

Note: Both 8210s need an attachment to another ELAN or LIS to enable IP routing (see discussion in 7.7.2.1, "Default Gateway and Dynamic Routing Protocols" on page 203).

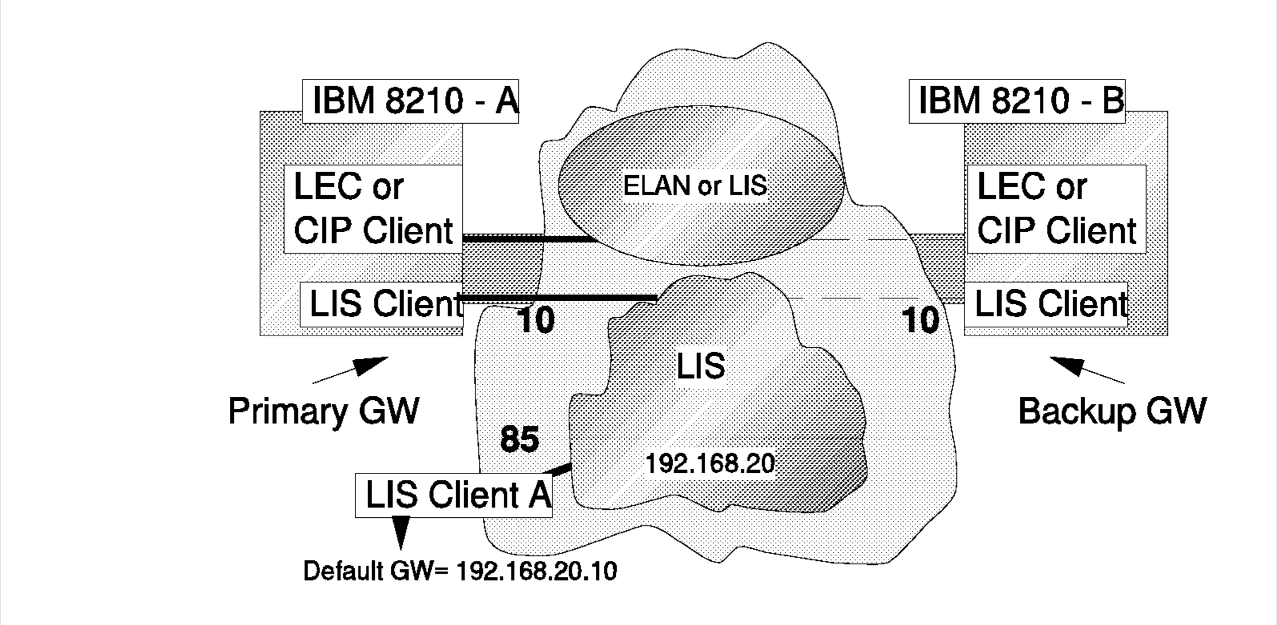


Figure 127. Default IP Gateway Redundancy - Using Classical IP

When the remote LIS client (for example, LIS client A in Figure 127) needs to send IP traffic to destinations outside of their local LIS, it requires service of the default gateway. Before it can use the default gateway, it requests the default gateway's ATM address from the ARP server, establishes a direct VCC and forwards the IP data.

Note: Because both 8210s use the same ATM address, only one default gateway can be registered at the ARP server.

The gateway backup mechanism is based on the fact that the gateway LIS client on both 8210s use the same IP address.

Important

Using the same IP address is only allowed when the 8210s do not join the LIS at the same time. Joining the LIS at the same time can be prevented by using the same ESIs for both LIS clients and attaching both 8210s to the same ATM switch. For details on defining ESIs, see 6.3.1, "LIS Client Using Dynamic SVCs" on page 157.

Because ATM switches do not allow duplicate ESIs to be registered, ESI registration of the backup 8210 will fail initially, but as the 8210 uses an ESI registration retry mechanism, it will be successful when the primary gateway fails. When its ESI registration has been completed, the backup gateway will register itself at the ARP server and become available for default routing.

Note: The IBM 8210 Nways MSS Server retries ESI registration for LIS clients every 15 seconds.

To benefit from the backup mechanism one should either use an ARP server that is outside of both 8210s or, much better, configure the default gateway LIS client also as an ARP server (see also 6.3.5, "Redundant ARP Server" on page 166).

7.7.2 Default Gateway Redundancy - LAN Emulation

Figure 128 depicts two 8210s on which an LE client has the same IP address (192.168.4.10) on both systems. Either 8210 can perform default IP routing for remote clients attached to the same ELAN.

Note: Both 8210s need an attachment to another ELAN or LIS to enable IP routing (see discussion in 7.7.2.1, "Default Gateway and Dynamic Routing Protocols" on page 203).

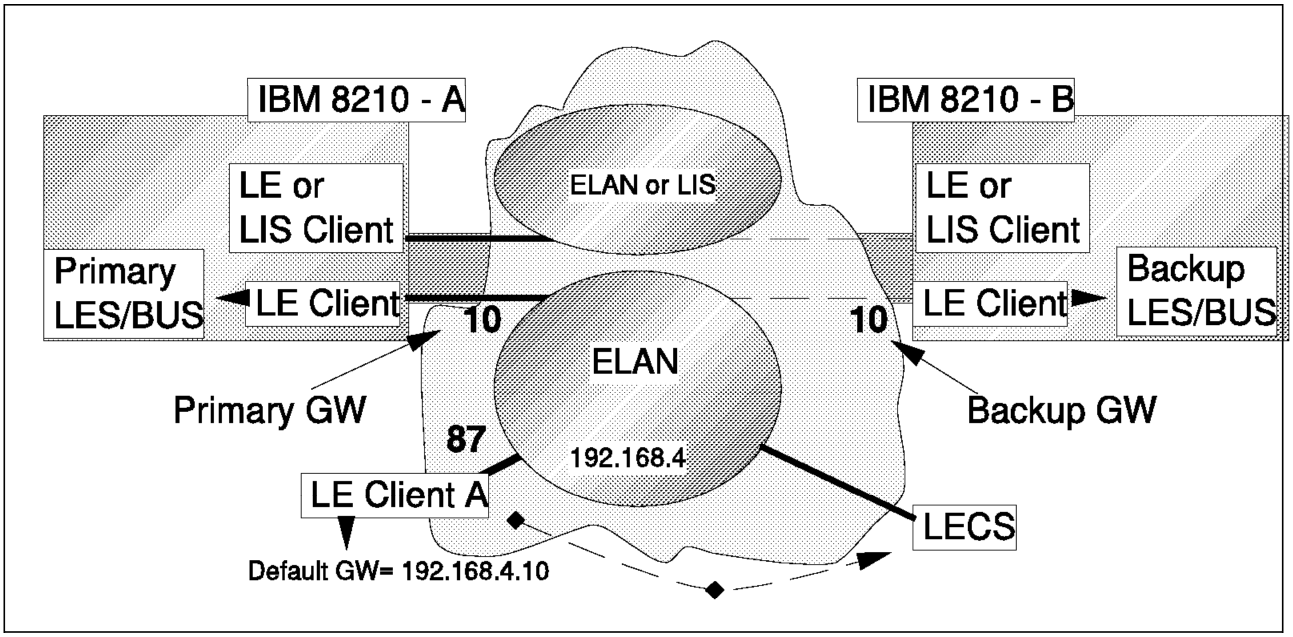


Figure 128. Default IP Gateway Redundancy - Using LAN Emulation

When the remote LE client (for example, LE client A in Figure 128) needs to send IP traffic to destinations outside of its local subnet, it requires service of the default gateway. Before it can use the default gateway, it broadcasts an IP ARP request on the ELAN, learns the MAC address associated with the default gateway LE client, establishes a data direct VCC and forwards the IP data.

To understand how backup is provided when the primary gateway fails, one needs to look at which 8210 is providing LES/BUS services for the ELAN.

Important

It is essential that all 8210-external LE clients attached to the ELAN use an LECS to learn the LES/BUS address.

Hereby, the LES on the primary gateway is defined as primary LES and the LES on the backup gateway as backup LES. As a result, external LE clients will only use the LES/BUS on the backup machine, when the LES on the primary machine fails. In contrast with the external LE clients, the default gateway LE clients should not use the LECS to learn the active LES address. Instead they must hard-code the ATM address of the LES that resides on the same 8210.

The backup mechanism is based on the fact that the backup gateway LE client will never be able to join the ELAN, unless the primary LES fails. When the primary LES fails, all clients have to join the ELAN again (after learning the backup LES address from the LECS) and use IP routing function of the default IP gateway LE client instead.

Important to realize in this backup mechanism is that:

1. There is no need to attach both 8210s to the same ATM switch, as the ATM addresses of the default gateway LE clients can be different.
2. The primary default gateway will take over, when the primary LES becomes available again.

7.7.2.1 Default Gateway and Dynamic Routing Protocols

The default gateway redundancy discussed in the previous section is most useful when the remote clients do not support any dynamic routing protocols. Especially when using Classical IP this is not unlikely as RIP is not supported.

As can be seen in Figure 127 on page 201 and Figure 128 on page 202, the 8210s need an additional ELAN or LIS attachment to be able to perform IP routing. On this ELAN/LIS attachment you have the choice of having both 8210s active at the same or, using a similar approach as discussed in the previous section, define the 8210s as mutual backup.

Important

Although the default gateway function can be active on a single 8210 only, other functions can be active on both 8210s at the same time.

If both 8210s are simultaneously active on the ELAN/LIS, different ATM and IP addresses have to be specified, and the use of a dynamic routing protocol is recommended.

Routing Protocol Limitations

During our testing we noticed that RIP on the backup gateway 8210 also broadcasts routing information for its inactive interfaces. Therefore, routing errors may result when using RIP as your dynamic protocol.

Chapter 8. MSS Server and IPX Routing

This chapter provides an overview of IPX routing and the implementation on the MSS Server. Included is an example of how to configure IPX routing using the Configuration Program.

Important

The MSS Server is capable of native ATM IPX routing using RFC 1483 encapsulation and IPX routing between ELANs. IPX routing using RFC 1483 is similar to the support for conventional IPX routers on X.25 or frame-relay networks while the ELAN routing functions are essentially the same as used by conventional IPX routers on legacy LANs.

Further information about IPX on the MSS Server can be found in the *Multiprotocol Switched Services (MSS) Server Command Line Interface Volume 2*.

8.1 IPX Addresses

IPX network interfaces have a unique 10-byte IPX address. The IPX address is not configured at the network device as it is in IP. The IPX address is dynamically constructed from the 6-byte MAC address and the learned local 4-byte IPX network number.

In order for an IPX client to communicate with a server, the client must first obtain the server′s IPX address, and possibly (IPX over ELAN only), the router′s MAC address which provides connectivity to the server′s IPX network.

8.2 RIP Overview

IPX uses the routing information protocol (RIP) to exchange routing information on a NetWare internetwork. NetWare routers use RIP to create and maintain a routing table. The routing table contains information on the networks reachable through that router.

Note: The RIP protocol used by NetWare is different from that used by IP.

The routing table will contain the following information for each network:

- Network number

  The *network number* is the number of the IPX network that the table entry relates to.

- Number of hops to the network

  The *number of hops* indicates the number of routers that must be crossed to reach the network.

- Number of ticks to the network

  The *number of ticks* is an estimate of the time required to reach the network (where one tick is approximately 1/18th of a second).

- Interface through which the network is reached

The *interface* indicates through which of the router′s interfaces this network is reachable.

- Address of the next hop router

  The *address of the next hop* indicates the address of the router that is the next hop to the indicated network. The MSS Server implementation puts the network/node of the directly-connected interface in the table.

- Aging timer

  The *aging timer* is used to ensure that the information kept is current.

To build and maintain the routing tables, RIP packets are exchanged between the routers. When a router is first initialized, it will place the network numbers of its directly connected networks into its routing table. It will then broadcast a RIP packet on each of its directly connected networks to inform routers on those networks of the networks it is making available.

The router will then broadcast a RIP request on each directly connected network, requesting information about all other networks available. All routers on those directly connected networks respond with information from their routing tables. The router then adds this information to its routing table and broadcasts a new RIP packet to the directly connected networks, containing the newly learned information.

Each router uses a *split-horizon* algorithm which prevents it from sending information about routes it learned back to the network it learned those routes from.

The algorithm ensures that when a router broadcasts to a locally connected network, it doesn′t include any information about other networks that it received from that locally connected network.

The algorithm also ensures that a router does not include information about the network it is broadcasting on.

Important

If MSS is providing intermediate routing using RFC 1483 between VCCs on a non-fully meshed ATM interface, IPX split-horizon must be disabled on that ATM interface so that RIP routes and SAP services learned via one VCC are advertised on the other VCCs on that same interface. See also 8.4, “Overview of RFC 1483 Support for IPX Routing” on page 211.

As ELANs are per default fully-meshed, IPX split-horizon can be enabled on ELAN connections.

The router will now periodically broadcast RIP packets containing all the information in its routing table to its directly connected networks. A configurable parameter, called the *RIP update interval*, determines the interval between these broadcasts.

The router will also broadcast a RIP packet when it detects a network change, such as one of its directly connected networks going down. In this instance, the packet informs the other routers that all networks accessible through the failed network are now unreachable. It does this by setting the hop count for those networks to 16. (A network is considered unreachable if the number of hops is 16.)

When a router receives a RIP packet indicating a network change, it will update its routing table and inform other routers on its directly connected networks of the changes.

A RIP packet will also be sent, informing other routers that a network is unreachable, when the aging timer for the network expires. The timer expires if the router hasn′t received any information for that network in a specified time. This mechanism allows the routers to handle the sudden loss of another router.

By using these RIP requests and responses, all routers on the internetwork are able to stay in synchronization.

8.2.1 RIP Packet Format

The RIP packet format is shown in Figure 129. The packet is contained within the data portion of an IPX packet. RIP packets are defined within the IPX header as Packet Type 1 and Socket Number X′453′.

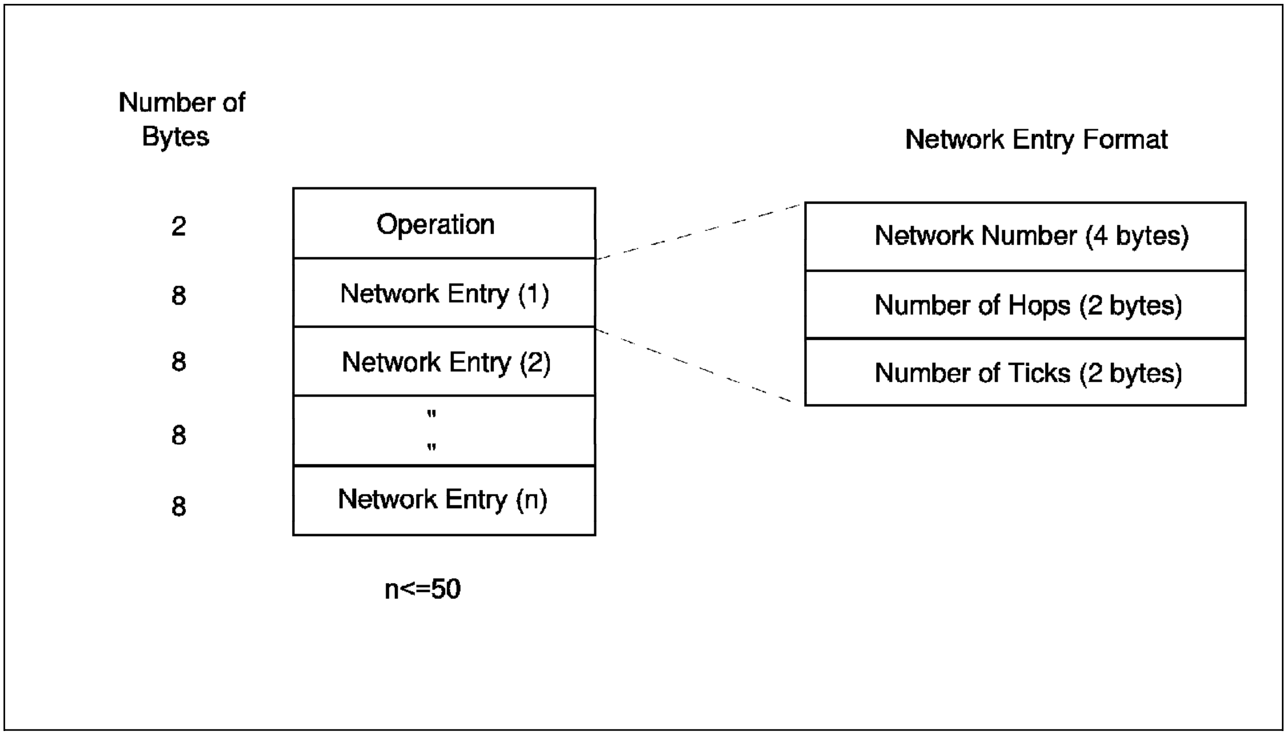


Figure 129. RIP Packet Format

Operation: This field indicates whether the packet is a request or a response. If it is a request, the field contains 1, and if it is a response, the field contains 2. A request can be for a specific network or a general request, such as when the router first starts and needs to learn the existing networks. For a general request, the network number field will contain X′FFFFFFFF′.

A response is used to reply to a request and also to send informational packets, such as those sent periodically.

Network Entry: There can be up to 50 network entries in a RIP packet and each network entry contains the network number, number of hops, and number of ticks. If there is information for more than 50 networks to be broadcast, multiple RIP packets are used.

8.2.2 Route Selection

On large internetworks, there may be multiple routes to a single network. It is the responsibility of the router to select the best route to that network. The following is the criteria used for the best route selection:

1. Select the route that requires the lowest number of ticks.
2. If multiple routes exist with the number of ticks equal, select the route that also has the lowest number of hops.
3. If multiple routes exist with both ticks and hops equal, the router is free to select any of these routes as the best route.

The routing table might contain a list of routes to each network in case the best route goes down. Alternatively, it might contain alternate routes only if those routes require the same amount of ticks as the best route. Another option would be for the routing table to contain only the best route.

The router is free to choose if it will store only the best route, all routes, or some alternate routes. When sending RIP responses, however, routers should only include the best route information.

> **MSS**
>
> The MSS Server can be configured to store between 1 and 64 equal-cost routes to each destination IPX network. Equal-cost means that the routes have the same ticks and hops. Only one route is used at a time; the others are kept as alternates in case the route being used is lost or ages out.

8.3 SAP Overview

IPX uses the service advertising protocol (SAP) to exchange information about available services on a NetWare network. Service-providing nodes, such as file servers and print servers, use SAP to advertise their services and addresses. Routers use SAP to create and maintain a database of services available on the internetwork. This allows clients to determine what services are available and to obtain the addresses of the servers where they can access those services. This is important, as clients cannot access a file server unless they know the server′s address.

The database maintained by the router is commonly called the Server Information Table. It contains the following information for each server:

- Server name

  The *server name* is the name assigned to the server.

- Server address

  The *server address* is a combination of the network address, the node address and the socket address for the server.

- Server type

The *server type* indicates the type of server this entry represents.

- Hops to server

  The *hops to server* indicates the number of routers that must be crossed to reach the server.

- Interface through which the information was received

  The *interface* indicates through which of the router′s interfaces this SAP information was received.

- Aging timer

  The *aging timer* is used to ensure that the information kept is current.

To build and maintain the server information tables, SAP packets are exchanged between routers and servers.

When a router is first initialized, it broadcasts a general SAP request on all of its directly connected networks, requesting information about all available servers on the internetwork. SAP agents on those directly connected networks will respond with information about the servers available, and the router will place that information in its server information table. The router then sends a SAP packet to its directly connected networks, containing the newly learned information.

The router will now periodically broadcast SAP packets containing all the information in its server information table to its directly connected networks. A configurable parameter, called the *SAP update interval*, determines the interval between these broadcasts.

The router will also broadcast a SAP packet when it detects a network change, such as one of its directly connected networks going down. In this instance, the packet informs the other routers that all servers accessible through the failed network are now unreachable. It does this by setting the hop count for those servers to 16.

When a router receives a SAP packet indicating a network change, it will update its server information table and inform other routers on its directly connected networks of the changes.

A SAP packet will also be sent, informing other routers that a server is unreachable, when the aging timer for the server expires. The timer expires if the router hasn′t received any information for that server in a specified time. This mechanism allows the routers to handle the sudden loss of another router or server.

By using these SAP requests and responses, all routers and servers on the internetwork are able to stay in synchronization.

8.3.1 SAP Packet Format

The SAP packet format is shown in Figure 130 on page 210. The packet is contained within the data portion of an IPX packet. SAP packets are defined within the IPX header as Packet Type 4 and Socket Number X′452′.

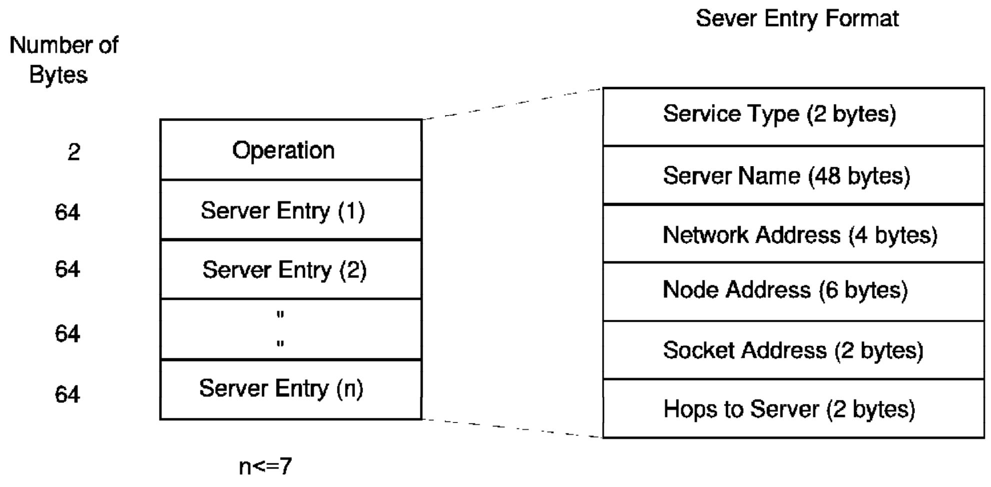


Figure 130. SAP Packet Format

Operation: This field indicates the type of operation being performed by this SAP packet and can be one of the following values:

- Request
- Response
- Get Nearest Server Request
- Get Nearest Server Response

The Get Nearest Server Request and Get Nearest Server Response are discussed in 8.3.2, “Get Nearest Server Request Handling” on page 211.

For all SAP requests, the Operations field is followed by a single Service Type field of two bytes and nothing else.

All other fields shown are for response packets only.

Service Type: This field identifies the type of service being provided by the server. Some of the well known server types are:

- X′0003′ Print Queue
- X′0004′ File Server
- X′0007′ Print Server
- X′0009′ Archive Server

Server Name: This field contains a 48-byte name that is assigned to a server. Typically, server names are not 48 characters long, but SAP response packets always include the full 48 bytes. The bytes after the end of the actual server name (which is marked by the ASCII null character) are undefined and may contain anything.

Network Address: This field contains the server′s network address.

Node Address: This field contains the server′s node address.

Socket Address: This field contains the socket number on which the server will receive service requests.

Hops to Server: This field indicates the number of routers that must be crossed to reach the server.

8.3.2 Get Nearest Server Request Handling

When a router receives a Get Nearest Server Request, it should first check to see if any servers of the type requested reside on the same network as the requesting station. If that is the case, then the router should not respond to the Get Nearest Server Request as the server will do so itself. If no server exists locally to the requester, then the router should respond with the nearest server of the type requested. The criteria used to determine the nearest server are:

1. Select the server which has the best route as determined by the routing table.
2. If multiple servers exist with equally good routes, select the server that also has the least number of hops as determined from the server information table.

8.4 Overview of RFC 1483 Support for IPX Routing

RFC 1483 specifies encapsulation methods for carrying bridged and routed protocols over AAL-5 ATM connections. Using RFC 1483 the MSS Server supports connections to IPX routers over the native ATM interface.

IPX routers use RIP and SAP to propagate routing and server information tables. On legacy or emulated LANs these protocols use broadcast frames to propagate information. The MSS Server will also propagate this information to/from all RFC 1483 connections with adjacent routers.

The RFC connection with other routers can be done with PVCs or SVCs. All routers must be on the same IPX network and the RFC 1483 VCCs to other routers may not be shared with other protocols.

The destination IPX host numbers may be specified during VCC configuration or learned dynamically via InATMARP (if the destination router supports this method). If the remote host number is specified during VCC configuration, it does not age out of the InATMARP address mapping table. If it is learned dynamically via InATMARP, it will age and InATMARP is responsible for refreshing the table entry periodically. InATMARP has no affect on the routing tables.

Care should be taken with the use of IPX split-horizon. IPX routing using RFC 1483 is similar to the support for conventional IPX routers on X.25 or frame-relay networks. If MSS is providing intermediate routing between VCCs on a non-fully meshed ATM interface, IPX split-horizon must be disabled on that ATM interface so that RIP routes and SAP services learned via one VCC are advertised on the other VCCs on that same interface.

The host number of a local IPX protocol address is automatically set to the ESI component of the ATM associated address.

8.5 IPX on the MSS Server

IPX support on the MSS Server is compatible with all previous Novell Netware version environments. It is also compatible with the bridging function in a Netware file server and supports the Novell NetBIOS emulator.

IPX is supported over token-ring and Ethernet emulated LANs and also over native ATM (RCF 1483 encapsulation).

For Ethernet emulated LANs, the supported frame types (one per interface) are:

- Ethernet_8023 (uses 802.3) default type
- Ethernet_II (DIX V2)
- Ethernet_8022 (uses 802.2 (SAP E0))
- Ethernet_SNAP (uses SNAP (000000 8137))

For token-ring emulated LANs, the supported frame types (one per interface) are:

- Token-ring MSB (uses 802.2 (SAP E0) with non-canonical IPX address) default type
- Token-ring LSB (uses 802.2 (SAP E0) with canonical IPX address)
- Token-ring_SNAP (uses SNAP (000000 8137) with non-canonical IPX address)
- Token-ring_SNAP (uses SNAP (000000 8137) with canonical IPX address)

MSS Server also implements a series of filters:

- Global IPX filter (access control)

  This filter is applied to all interfaces and is used to prevent routing based on the destination/source network, node or socket number range. It operates on IPX packet header only in input direction for local and routed packets.

  Note: The use of a global IPX filter is enabled when selecting Enable access control in Figure 143 on page 225.

- Global SAP filter

  This filter is applied to all interfaces and is used to prevent SAP information based on the number of hops, the service type or the service name. It operates on SAP general response entries in input direction for periodic and triggered SAPs broadcasts.

- RIP router filter (interface filter)

  This filter operates on source address of RIP responses only in input direction. It uses the router node/mask as parameter and can be used to divide one IPX network into several distinct networks.

  Note: The use of a RIP router filter is enabled when selecting Enable RIP router filter lists in Figure 143 on page 225.

- RIP filter (interface filter)

  This filter operates on RIP requests/responses in input and/or output directions. It uses network number range as parameter and can be used to control the extent to which the routing information about a selected network is disseminated.

  Note: The use of RIP interface filters is enabled when selecting Enable RIP filter lists in Figure 143 on page 225.

- SAP filter (interface filter)

  The interface SAP filter provide more extensive filtering control than the global SAP filter. It provides a flexible control of hop count, and provides wildcard support in the server name. Separate input and output interface filters can be used. The global and the interface filters are mutually exclusive.

  Note: The use of SAP interface filters is enabled when selecting Enable SAP filter lists in Figure 143 on page 225.

- IPX filter (interface filter)

  The interface IPX filter provide more extensive filtering control than the global IPX filter. The interface IPX filter provides filtering based on hop count, packet type and allows pattern matching in the source and destination node number and ranges in the source and destination network numbers. Separate input and output interface filters can be used. The global and the interface filters are mutually exclusive.

  Note: The use of IPX interface filters is enabled when selecting Enable IPX filter lists in Figure 143 on page 225.

For configuration details using the Configuration Program, see 8.6, “Configuration Details” on page 214.

8.5.1 IPX Configuration Commands

This section summarizes the IPX configuration commands available within the Config process of the MSS Server. To access the commands, you need to issue the `Config>`**`protocol ipx`** command.

Table 8 shows the commands available.

The *Multiprotocol Switched Services (MSS) Server Command Line Interface Volume 2* covers these commands in detail.

Table 8. IPX Configuration Commands

Command	Function
? (Help)	Lists all the IPX configuration commands, or lists the options associated with specific commands.
Add	Adds global IPX filters and global SAP filters.
Delete	Deletes global IPX filters and global SAP filters.
Disable	Disables specific IPX interfaces or globally disables IPX.
Enable	Enables specific IPX interfaces or globally enables IPX.
Exit	Exits the IPX configuration environment and returns to the CONFIG environment.
Filter-lists	Enters the prompt for the configuration of the interface filters. Table 9 on page 214 shows the commands for this prompt.
Frame	Specifies the data link format for Ethernet and token-ring interfaces.
List	Displays the current IPX configuration.
Move	Changes the line numbers set when adding access control.
Set	Sets the host number, network number, maximum RIP and SAP table sizes, global IPX and SAP filters, maximum service parameters, local and remote cache size, router name, RIP/SAP update intervals, and split-horizon usage.

Table 9. IPX Filter Configuration Commands

Command	**Function**
? (Help)	Lists all the IPX filter configuration commands, or lists the options associated with specific commands.
Attach	Attaches a specified filter-list to a specified filter.
Create	Creates a filter or a filter-list.
Default	Sets the default action of a filter to include or exclude.
Detach	Detaches a filter-list from a filter.
Disable	Disables filtering.
Enable	Enables filtering.
Exit	Exits the IPX filter configuration environment and returns to the general IPX configuration prompt.
List	Displays the current filter configuration
Move	Records filter-lists attached to a filter.
Set-cache	Sets the caching size for a specific filter.
Update	Accesses the `IPX` *type*`-LIST` *list-name* `Config>` prompt.

8.6 Configuration Details

As an example how to configure IPX routing functions using the MSS server, we set up an IPX routing scenario between three IPX networks. Two networks are represented by emulated LANs, a token-ring and an Ethernet emulated LAN, respectively. The third network is using RFC 1483 encapsulation on ATM interface 0 (port 1). The token-ring ELAN attachment is using IPX network number 1, the Ethernet attachment IPX network number 2, and the RFC 1483 interface IPX network number 3.

The LE clients connect directly to the LES without using an LECS. To simplify the definitions we assume that the LES for both token-ring and Ethernet ELANs are external. On the RFC 1483 network interface we connect to two remote routers, one via a PVC and one via an SVC.

Figure 131 on page 215 shows this configuration as well as the most important parameters.

To achieve this scenario the following configuration steps are required:

1 Define the ATM port(s)

All traffic of each MSS Server′s IPX network interface needs to be associated with a specific physical port. Per physical port you can define multiple (logical) IPX network interfaces; one RFC 1483 interface and multiple ELAN interfaces.

During configuration of the ATM port you have to configure:

a. The ATM port attributes

   For configuration details see Chapter 3, “MSS Server and ATM Ports” on page 53.

b. The end system identifiers (ESIs) associated with the ATM port

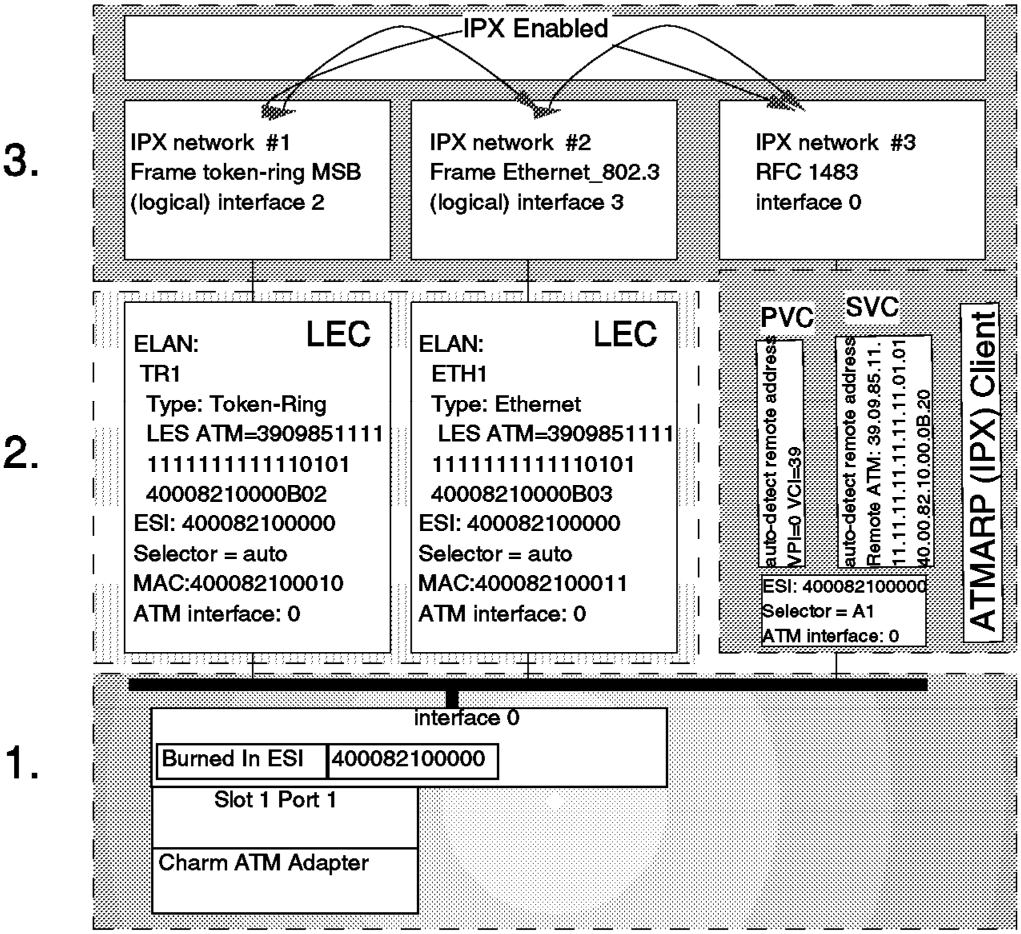


Figure 131. An Example of IPX Routing

During the definition of the LE clients and RFC 1483 VCCs you specify that either the burned-in end system identifier (ESI) or a user-defined ESI must be used. To simplify troubleshooting it is recommended that you use a locally administered ESI. For native ATM IPX connections, the ESI is used as a MAC address to comprise the IPX address (see 8.1, “IPX Addresses” on page 205).

When using RFC 1483 SVCs, a locally administered ESI simplifies the definition of the ATM addresses on remote IPX routers.

ESIs are administered per ATM port. A separate value has to be defined per ATM port. All LE clients and RFC 1483 IPX VCCs using a particular port can use the same ESI. Make sure that when defining multiple 8210s, the user-defined ESIs are unique.

Configuration of each ATM port on which IPX functions are required is needed. In our example all traffic is routed on the same physical interface. For the definitions entered, see 8.6.1, “ATM Port Definition” on page 216.

2 LEC configuration

IPX traffic may run on top of the LE client function provided by the MSS Server. For each IPX network interface define a separate LE client. For configuration details, see 8.6.2, “LE Client Definitions” on page 218.

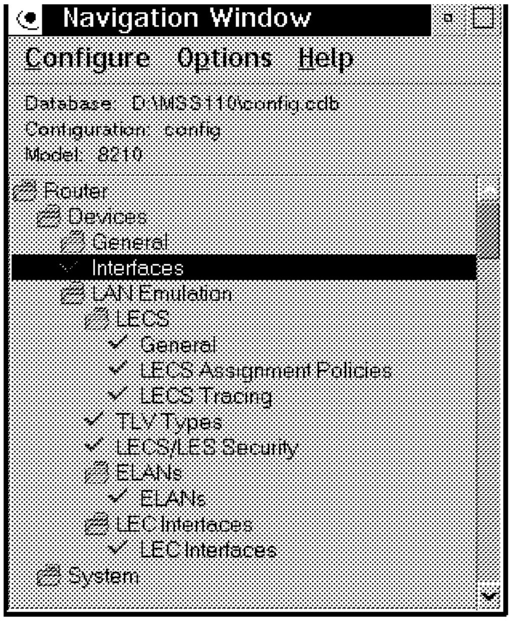
Note: As the MSS Server is routing, LE clients may be dissimilar (for example, token-ring and Ethernet).

3 IPX configuration

IPX needs to be enabled on box-level, and, for the relevant interfaces, on interface level. For each IPX interface set the proper IPX network parameters. For the RFC 1483 VCCs PVC and SVC definitions need to be added.

8.6.1 ATM Port Definition

To start the ATM interface configuration select Interfaces in the General folder in the Navigation Windows as shown in Figure 132.



Navigation Window

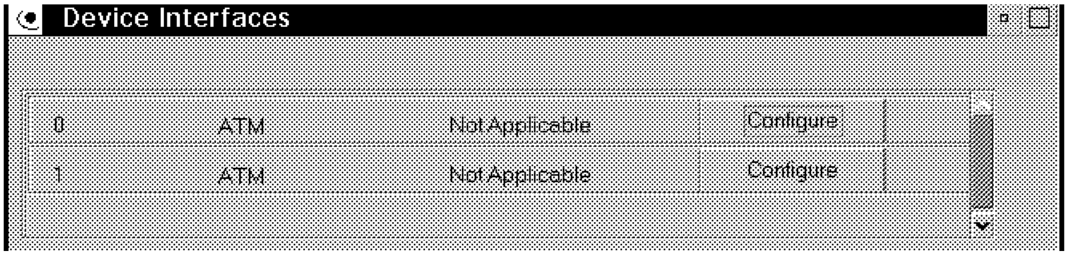
Configure Options Help

Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

- Router
  - Devices
    - General
    - Interfaces
    - LAN Emulation
      - LECS
        - General
        - LECS Assignment Policies
        - LECS Tracing
      - TLV Types
      - LECS/LES Security
      - ELANs
        - ELANs
      - LEC Interfaces
        - LEC Interfaces
  - System

Figure 132. Configure ATM Interface

Figure 133 will appear.



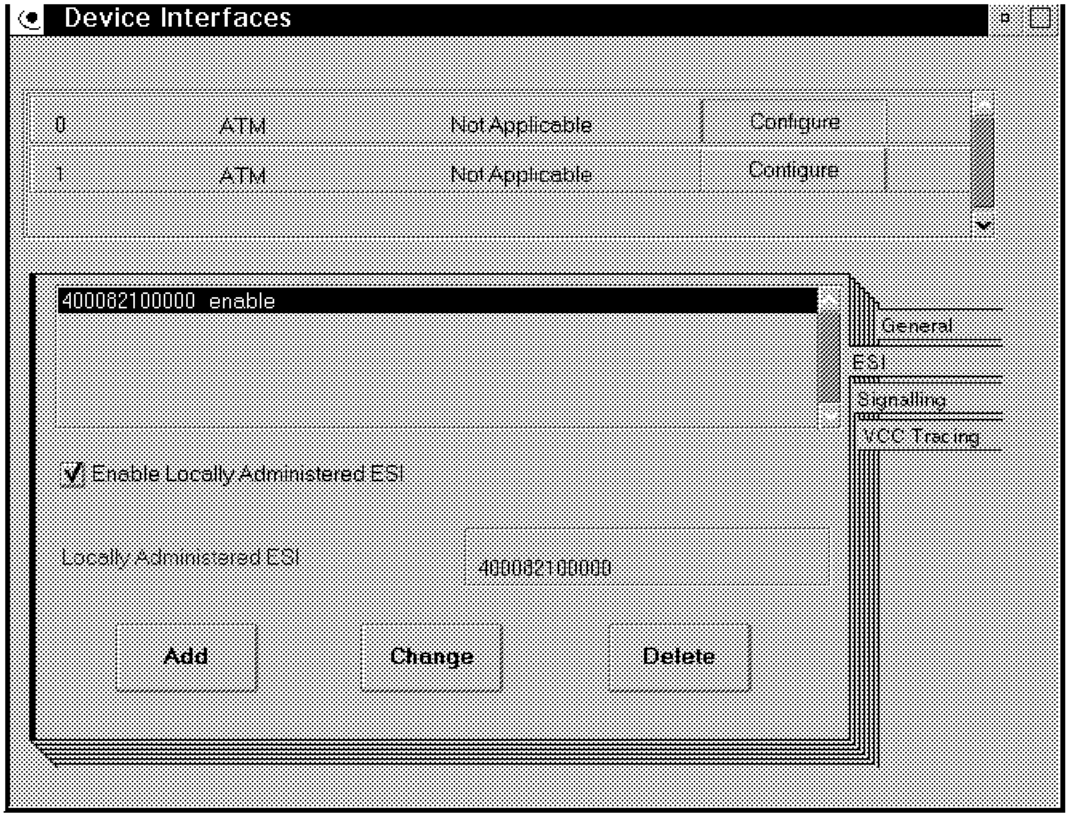
Device Interfaces

0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

Figure 133. ATM Interface

To simplify problem determination we decided to use a user-defined ESI for the LE clients and the RFC 1483 connections. If you want to use the *burned-in* ESI, you can skip this step.

To add an ESI click on the Configure button beside the interface you are planning to use (0 in our case). After selecting ESI, Figure 134 on page 217 appears.



0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

400082100000 enable

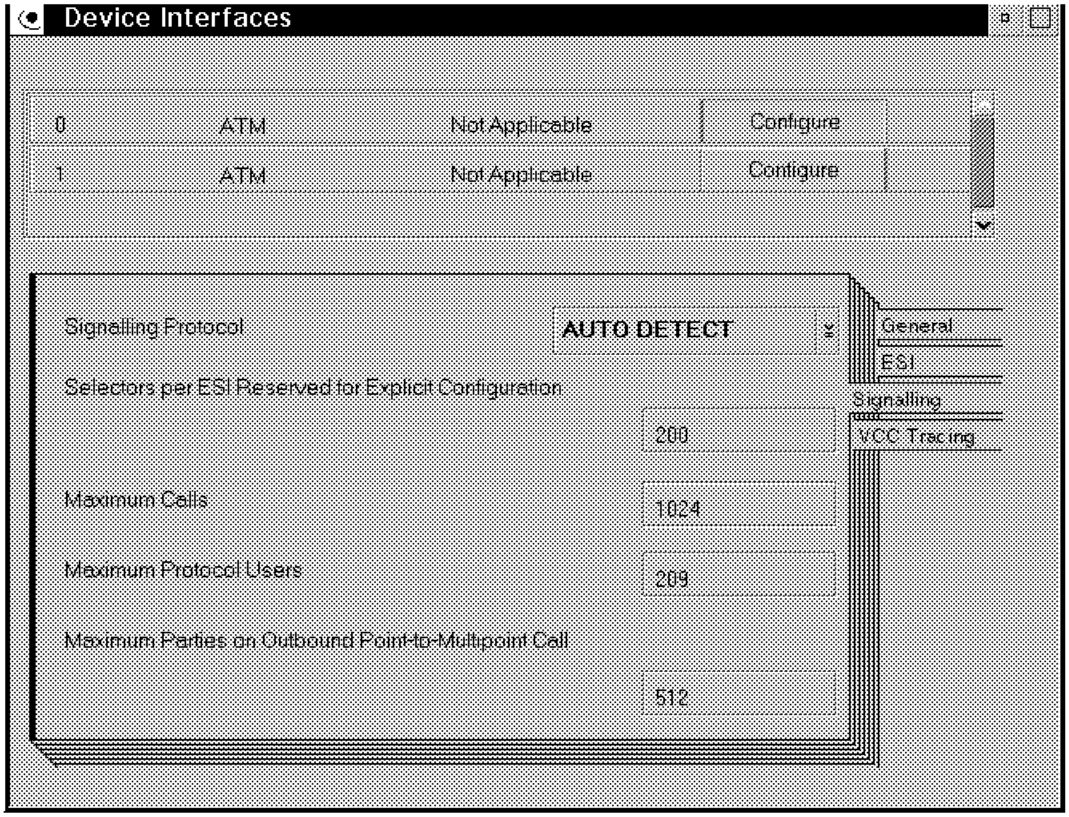
☑ Enable Locally Administered ESI

Locally Administered ESI: 400082100000

Figure 134. Adding an ESI

Here, you enter the locally-administrated ESI and then click on Add.

Next, configure the UNI version for this interface. After selecting Signalling, Figure 135 on page 218 appears.



0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

Signalling Protocol: AUTO DETECT

Selectors per ESI Reserved for Explicit Configuration: 200

Maximum Calls: 1024

Maximum Protocol Users: 209

Maximum Parties on Outbound Point-to-Multipoint Call: 512

Figure 135. Configuring the UNI Version

Select the appropriate UNI version (Signalling Protocol). If you are not sure about the UNI version that has been configured on the adjacent ATM switch port, choose the auto detect option. This ends the configuration of the ATM interface.

Note: If the UNI version is incorrect, the MSS Server may not work (see Chapter 3, “MSS Server and ATM Ports” on page 53).

8.6.2 LE Client Definitions

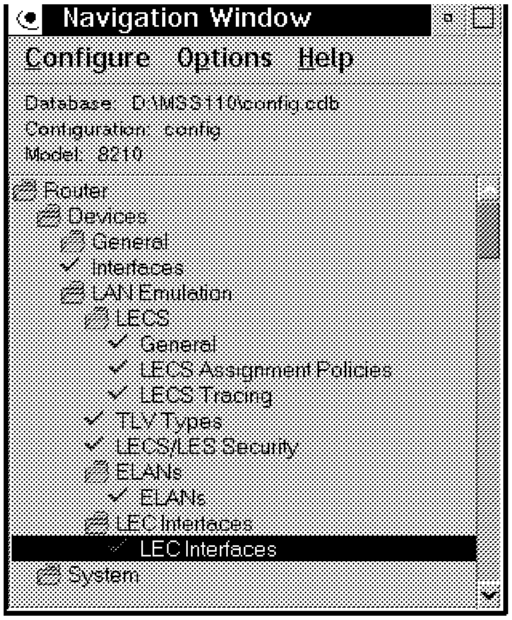
The next step during IPX configuration is to configure each of the LE clients, if any, on top of which IPX functions will be enabled. In our example we are using a token-ring and an Ethernet LE client.

Important

An LE client can be simultaneously used for IPX, IP, and bridging.

Note: Because the definitions are almost identical, we detail the configuration of the token-ring LE client only.

To start the LE client configuration, select LEC Interfaces in the Navigation Window, as shown in Figure 136 on page 219.

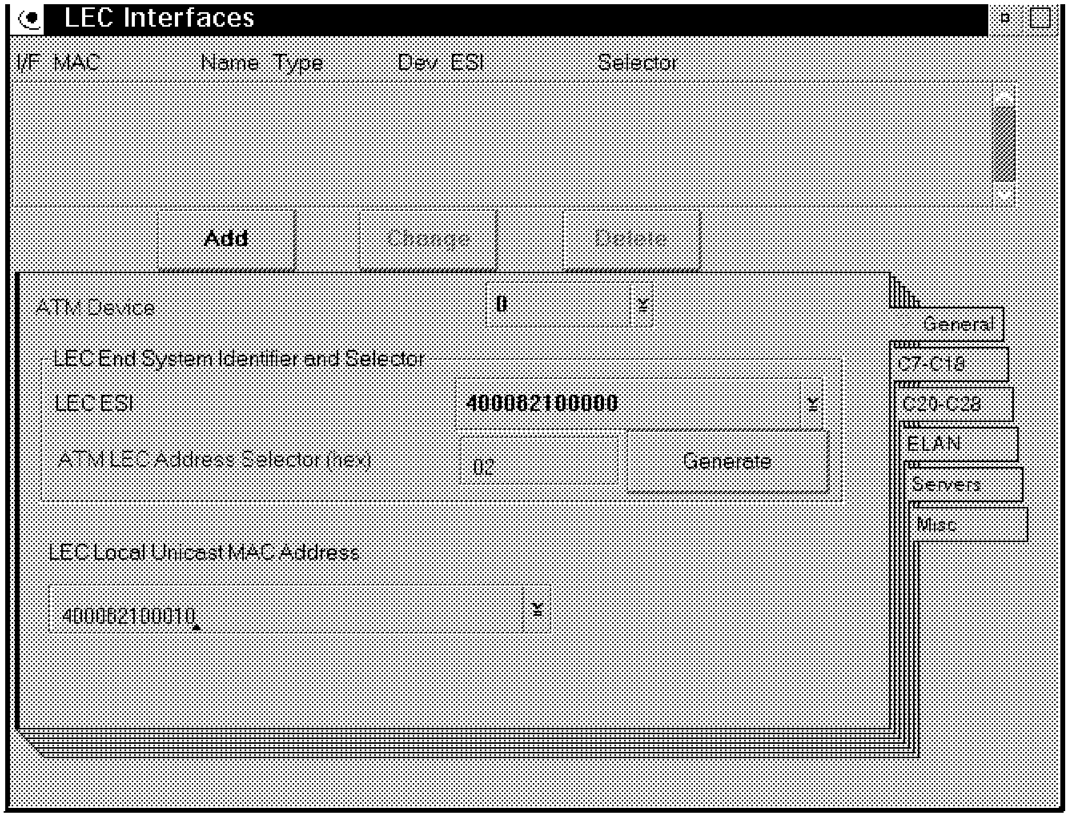


Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

- Router
  - Devices
    - General
    - ✓ Interfaces
    - LAN Emulation
      - LECS
        - ✓ General
        - ✓ LECS Assignment Policies
        - ✓ LECS Tracing
      - ✓ TLV Types
      - ✓ LECS/LES Security
      - ELANs
        - ✓ ELANs
      - LEC Interfaces
        - ✓ LEC Interfaces
  - System

Figure 136. Configure LEC

As a result Figure 137 will appear.



ATM Device: 0

LEC End System Identifier and Selector

LEC ESI: 400082100000

ATM LEC Address Selector (hex): 02

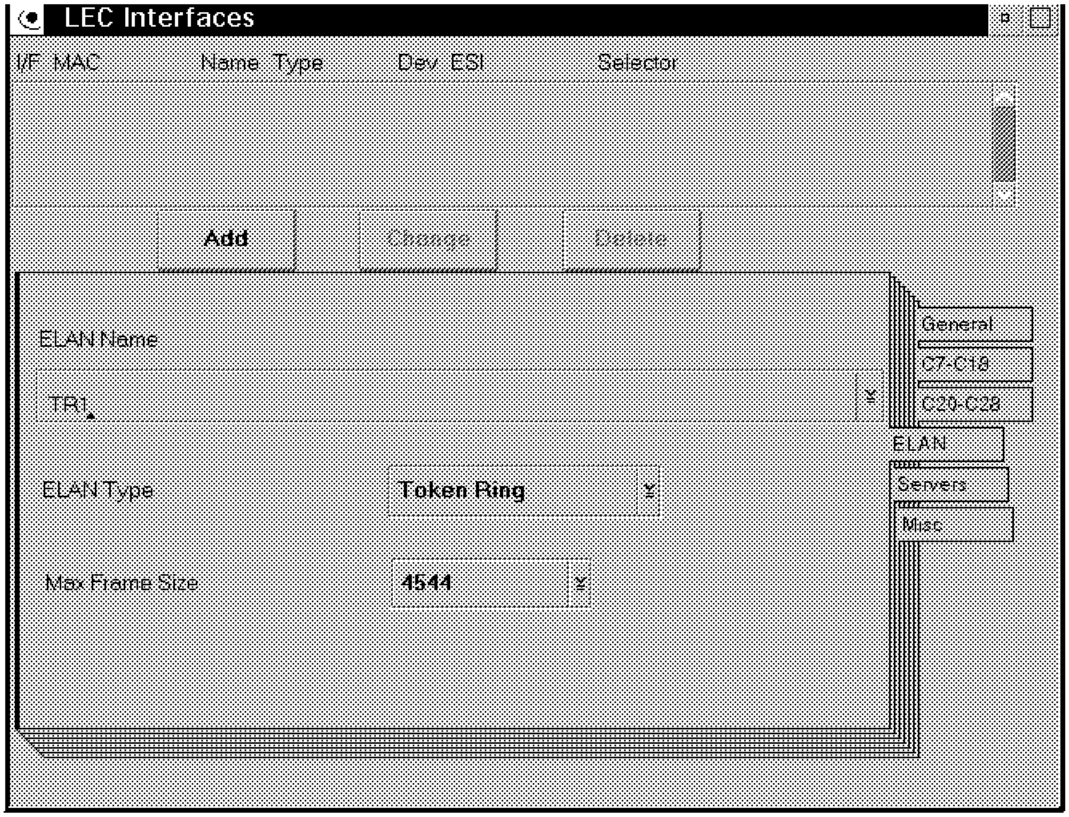
LEC Local Unicast MAC Address: 400082100010

Figure 137. Configure ESI, MAC, and Selector Byte

Here we select the ATM port (device) which will be used by the LE client, select our user-defined ESI, have the configurator generate a selector byte, and enter a MAC address. It is advised to use unique MAC addresses. The MAC-address together with the IPX network assigned to the LE client comprise the IPX address, see 8.1, “IPX Addresses” on page 205.

Selecting ELAN results in Figure 138. This configuration screen enables us to enter the ELAN type, the maximum frame size, and the ELAN name. Because our LE client connects directly to the LES, the ELAN name is optional.

Note: For documentation purposes it is advised that you use the same name as defined on the LES and LECS.



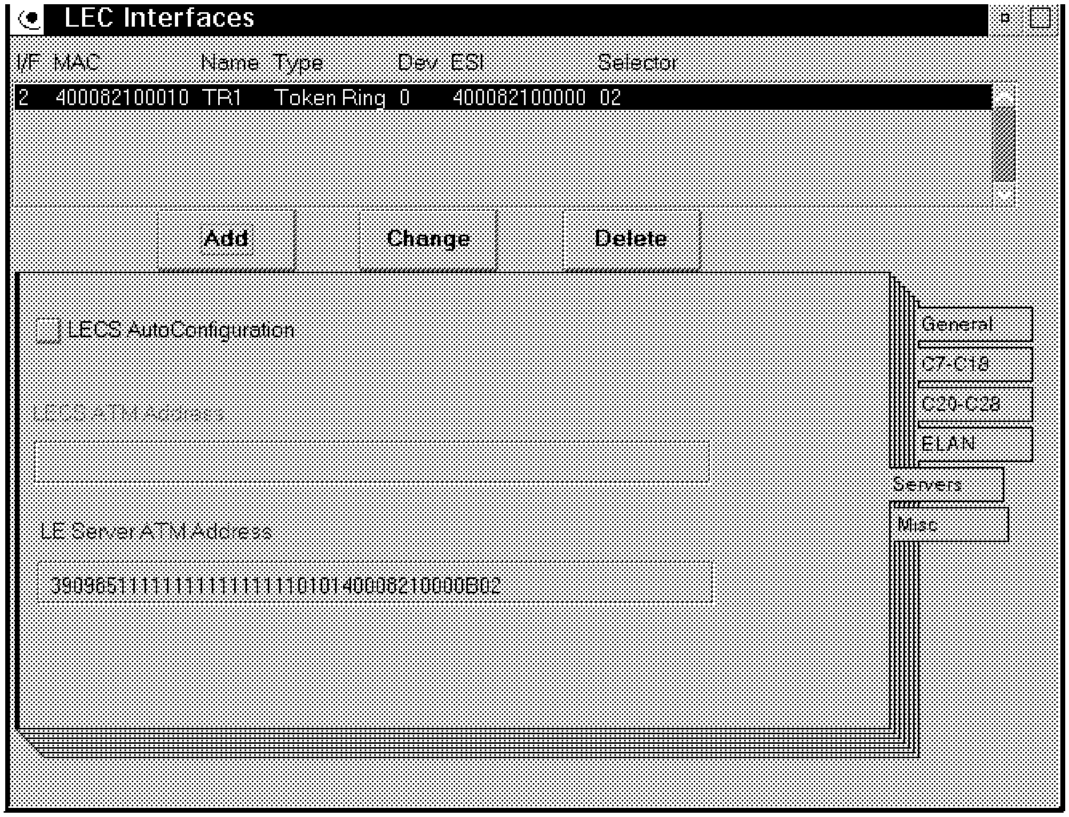
ELAN Name: TR1

ELAN Type: Token Ring

Max Frame Size: 4544

Figure 138. Configuring ELAN Type, Name, and Maximum Frame Size

The next step is to configure the ATM address of the LES of the token-ring ELAN to which the LE client connects. To do so we select Servers. As a result Figure 139 on page 221 appears.



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100010	TR1	Token Ring	0	400082100000	02

□ LECS AutoConfiguration

LECS ATM Address

LE Server ATM Address

3909851111111111111111010140008210000B02

Figure 139. Configure LES ATM Address

We configure the correct 20-byte LES ATM address and, because this LE client does not use LECS functions, disable LECS AutoConfiguration.

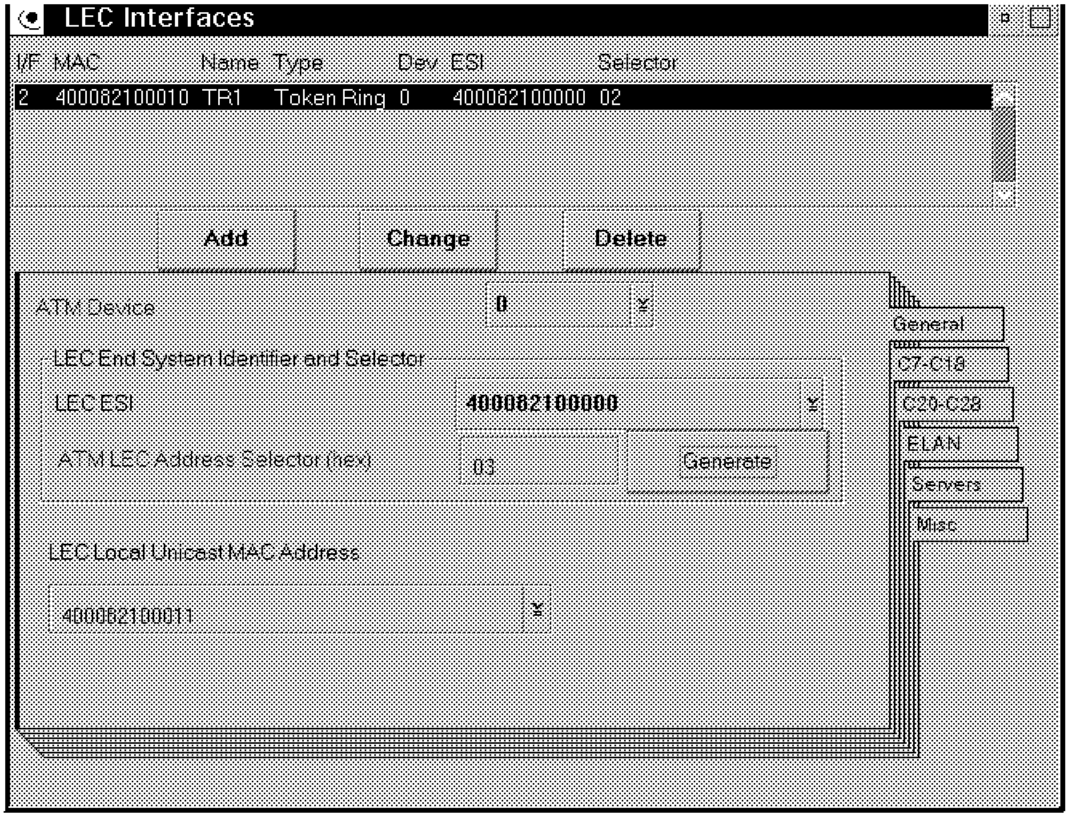
Selecting Add concludes the definition of the token-ring LE client.

As indicated in the top line, a token-ring LE client has been added to which logical interface (I/F) 2 has been assigned.

I/F Number

Remember the logical interface number assigned during LE client definition, as it should be used during setting the IPX parameters associated with this LE client.

The configuration of the Ethernet client can now be started. Figure 140 on page 222 appears after entering General.



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100010	TR1	Token Ring	0	400082100000	02

ATM Device: 0

LEC End System Identifier and Selector

LEC ESI: 400082100000

ATM LEC Address Selector (hex): 03

LEC Local Unicast MAC Address: 400082100011

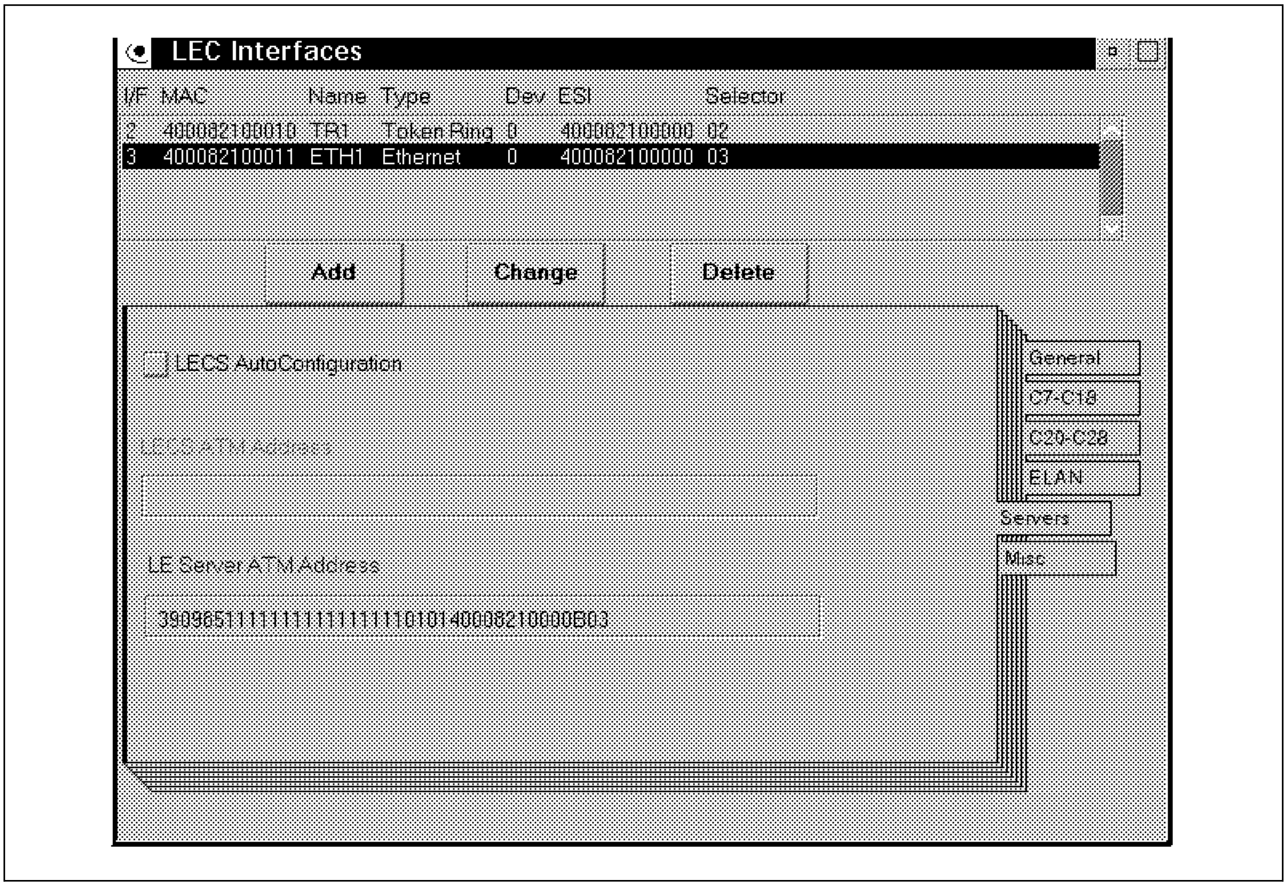
Figure 140. Configure LE Client

We use the same ESI for both LE clients and, using the configurator to generate one, a selector that is one higher than that used by the token-ring client. It is advised to use unique MAC addresses. The MAC-address together with the IPX network assigned to the LE client comprise the IPX address, see 8.1, “IPX Addresses” on page 205.

The procedure for configuring the Ethernet LE client is similar to defining the token-ring LE client and, therefore, not repeated. The main differences are that in the definition of the Ethernet LEC interface you have to make sure that the:

- Maximum frame size is small enough to avoid conflicts on Ethernet LAN segments (if any) that are bridged to the ELAN
- ELAN type is Ethernet
- LES ATM address points to the LES of the Ethernet ELAN

Note: The ELAN name is not important because the LE client connects directly to the LES. For documentation purposes, it is advised that you the same name as defined on the LES and LECS.



LEC Interfaces

```
I/F MAC          Name Type       Dev ESI          Selector
2   400082100010 TR1  Token Ring 0   400082100000 02
3   400082100011 ETH1 Ethernet   0   400082100000 03
```

□ LECS AutoConfiguration

LECS ATM Address

LE Server ATM Address

3909851111111111111111010140008210000B03

Figure 141. LES ATM Address Ethernet LE Client

Figure 141 results after both LE clients have been configured. Note that the interface number (I/F) assigned to the Ethernet LE client is 3.

This concludes the definition of the LE clients. The next step is the configuration of the IPX protocol.

8.6.3 IPX Configuration

The actual IPX configuration consists of the following steps:

- Define IPX global characteristics

  Enable IPX on box level, define RIP and SAP filters, etc.. For details see 8.6.3.1, “IPX - Global Configuration” on page 224.

- Define IPX characteristics for LE client interfaces

  This step is required for each LE client which is used as a IPX network interface. For details see 8.6.3.2, “IPX - LE Client Definitions” on page 226.

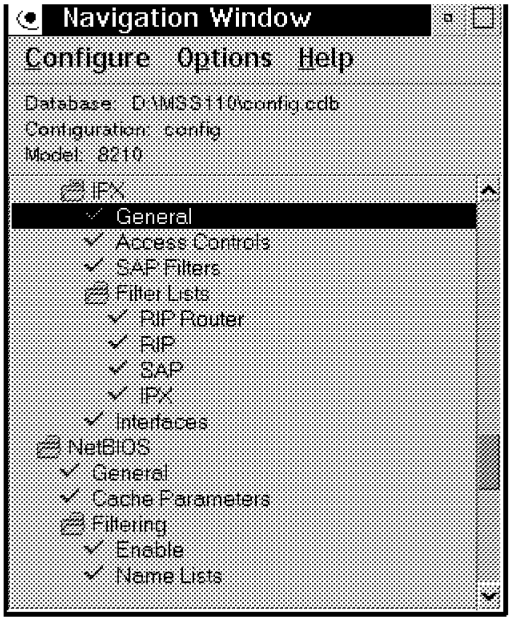
- Define IPX characteristics for RFC 1483 interfaces

  This step is required for each ATM port on which IPX routing using RFC 1483 is employed. For details see 8.6.3.3, “IPX - RFC 1483 Interface Definitions” on page 228.

  Note: Each ATM port may provide a single RFC 1483 network interface.

8.6.3.1 IPX - Global Configuration

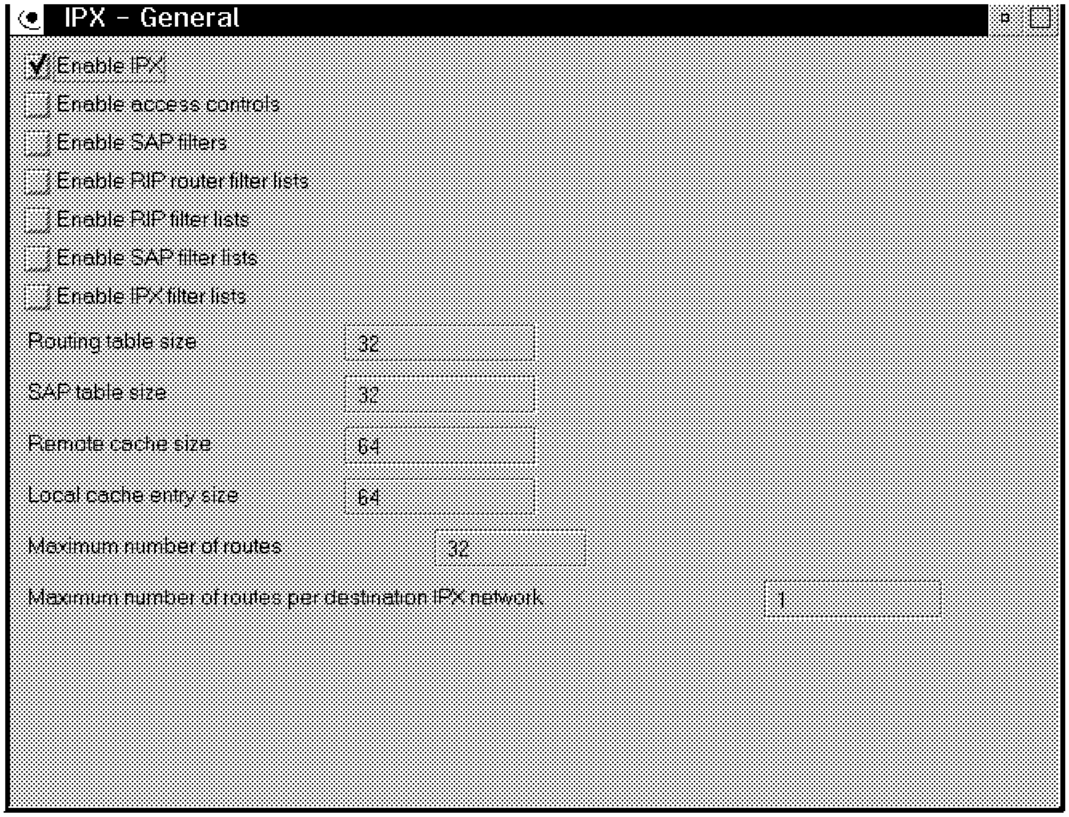
To start the actual IPX configuration we select General in the IPX folder on the Navigation Window, as shown in Figure 142.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

- IPX
  - ✓ General
  - ✓ Access Controls
  - ✓ SAP Filters
  - Filter Lists
    - ✓ RIP Router
    - ✓ RIP
    - ✓ SAP
    - ✓ IPX
  - ✓ Interfaces
- NetBIOS
  - ✓ General
  - ✓ Cache Parameters
  - Filtering
    - ✓ Enable
    - ✓ Name Lists

Figure 142. General IPX Configuration



IPX - General

☑ Enable IPX

□ Enable access controls

□ Enable SAP filters

□ Enable RIP router filter lists

□ Enable RIP filter lists

□ Enable SAP filter lists

□ Enable IPX filter lists

Routing table size 32

SAP table size 32

Remote cache size 64

Local cache entry size 64

Maximum number of routes 32

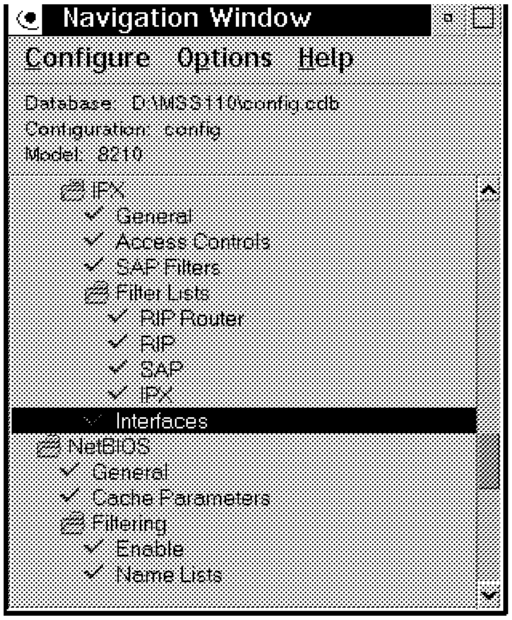
Maximum number of routes per destination IPX network 1

Figure 143. Enable IPX Globally

This results in the general IPX configuration screen shown in Figure 143.

Here we can enable IPX globally, enable SAP, RIP and IPX filters and change the global IPX parameters. In our case we only enabled IPX.

The next step is to configure IPX per interface. For that, select Interfaces in the IPX folder within the Navigation Window, as shown in Figure 144 on page 226. We will start with configuring the LE client interfaces, the RFC 1483 interface definitions are detailed in 8.6.3.3, “IPX - RFC 1483 Interface Definitions” on page 228.



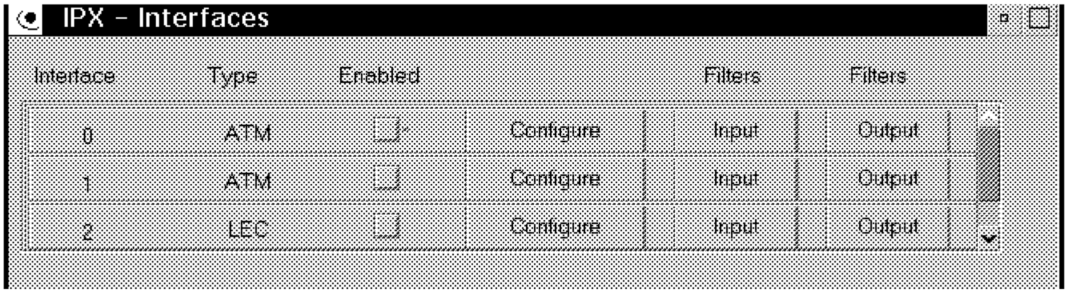
Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 144. Configuring IPX Interface

8.6.3.2 IPX - LE Client Definitions

After selecting Interfaces, Figure 145 appears in which all the configurable interfaces are listed. It includes the token-ring logical interface created during the previous LE client definitions.

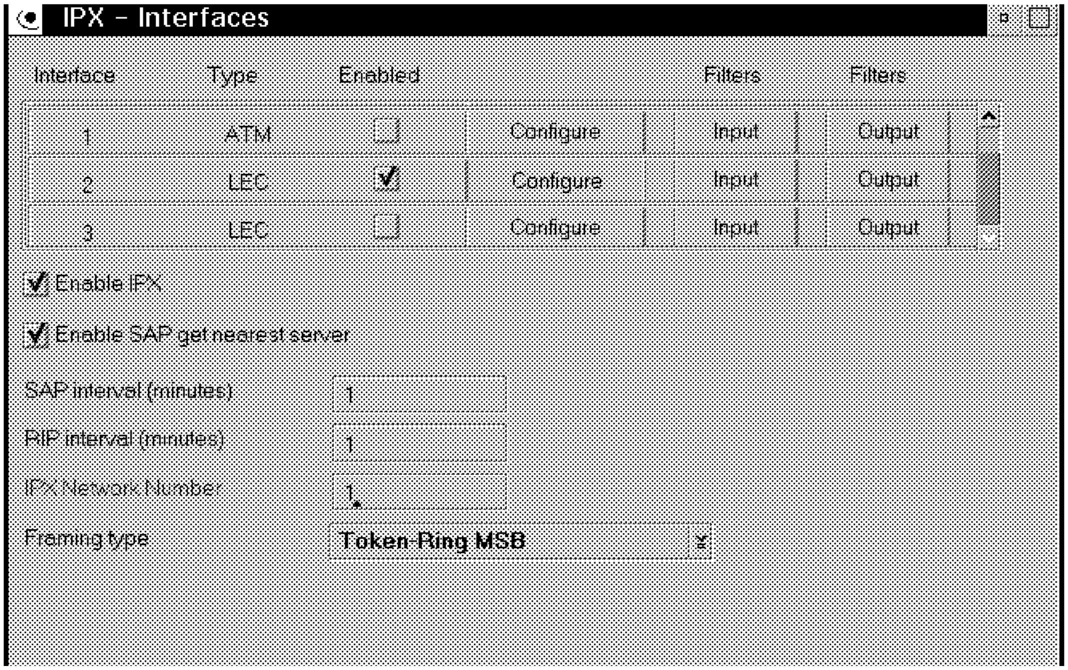
Note: To display the Ethernet LE client interface (number 3) one needs to scroll down.



Interface	Type	Enabled		Filters	Filters
0	ATM	☐	Configure	Input	Output
1	ATM	☐	Configure	Input	Output
2	LEC	☐	Configure	Input	Output

Figure 145. Configure IPX

Clicking on the Configure button for the token-ring logical interface (2) results in Figure 146 on page 227.



IPX - Interfaces

Interface	Type	Enabled		Filters	Filters
1	ATM	□	Configure	Input	Output
2	LEC	☑	Configure	Input	Output
3	LEC	□	Configure	Input	Output

☑ Enable IPX

☑ Enable SAP get nearest server

SAP interval (minutes): 1

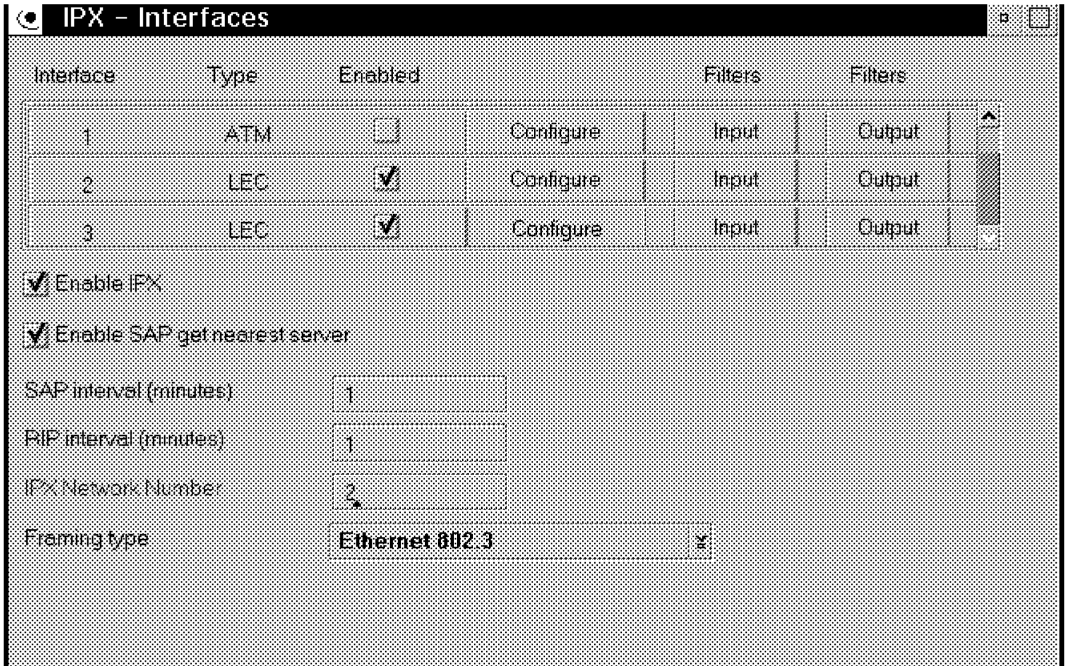
RIP interval (minutes): 1

IPX Network Number: 1

Framing type: Token-Ring MSB

Figure 146. Configure IPX Interfaces

Figure 146 enables use to activate IPX for the token-ring logical interface, specify the IPX network number, and indicate the frame type.



IPX - Interfaces

Interface	Type	Enabled		Filters	Filters
1	ATM	□	Configure	Input	Output
2	LEC	☑	Configure	Input	Output
3	LEC	☑	Configure	Input	Output

☑ Enable IPX

☑ Enable SAP get nearest server

SAP interval (minutes): 1

RIP interval (minutes): 1

IPX Network Number: 2

Framing type: Ethernet 802.3

Figure 147. Define IPX on LE Client

Figure 147 becomes available after selecting the Ethernet logical interface. We enabled IPX, specified the IPX network number, and indicated the appropriate frame type.

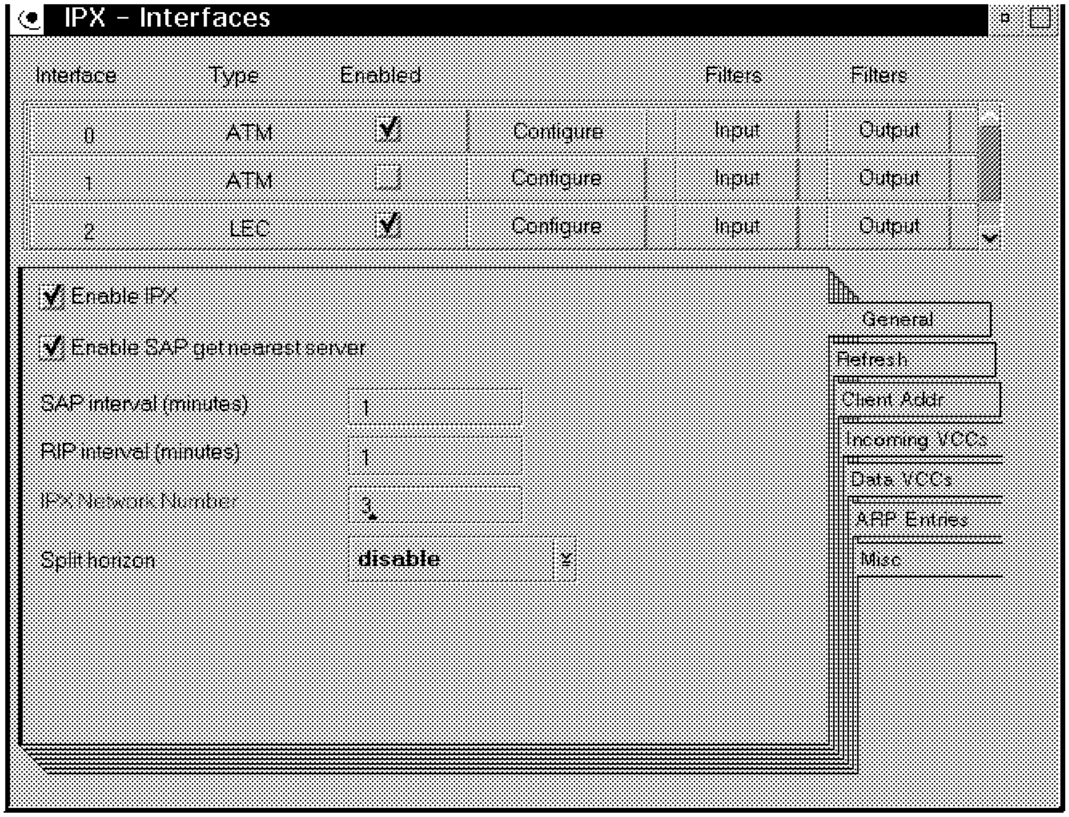
Notes:

1. Make sure the IPX network numbers differ on each of your logical interfaces.
2. Make sure the framing type is the same for all IPX stations that connect to the same ELAN.

After both LE clients have been configured for IPX, the RFC 1483 VCCs associated with interface 0 needs to be defined.

8.6.3.3 IPX - RFC 1483 Interface Definitions

To start the configuration of the RFC 1483 logical interface 0 associated with ATM interface 0 you have to go (select Interface in the IPX folder) to the IPX Interfaces screen depicted in Figure 145 on page 226. After an overview of the configurable interfaces appears select Configure for ATM interface 0 and select General on the right-hand side. Figure 148 will result.



Interface	Type	Enabled		Filters	Filters
0	ATM	☑	Configure	Input	Output
1	ATM	□	Configure	Input	Output
2	LEC	☑	Configure	Input	Output

☑ Enable IPX

☑ Enable SAP get nearest server

SAP interval (minutes) 1

RIP interval (minutes) 1

IPX Network Number 3

Split horizon **disable**

Figure 148. Configure IPX using RFC 1483

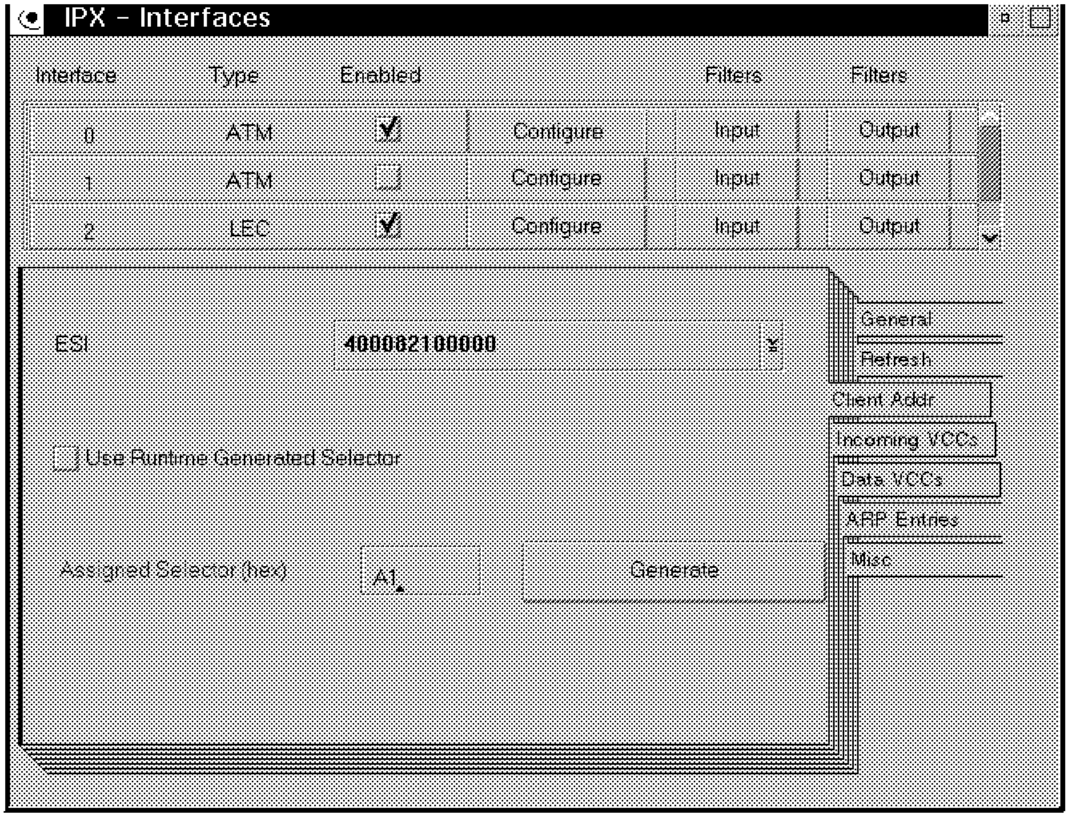
We enabled IPX and assigned network number 3 to this interface. Because we are not using a fully-meshed RFC 1483 network, we disabled split-horizon. For the other parameters we used the default values.

Next step is to define the MSS Server component that will be used to establish VCCs. For this purpose we selected Client Addr.

Important

Only one ATMARP (RFC 1483) IPX client can be defined per ATM port.

Figure 149 on page 229 appeared.



Interface	Type	Enabled		Filters	Filters
0	ATM	☑	Configure	Input	Output
1	ATM	□	Configure	Input	Output
2	LEC	☑	Configure	Input	Output

ESI 400082100000

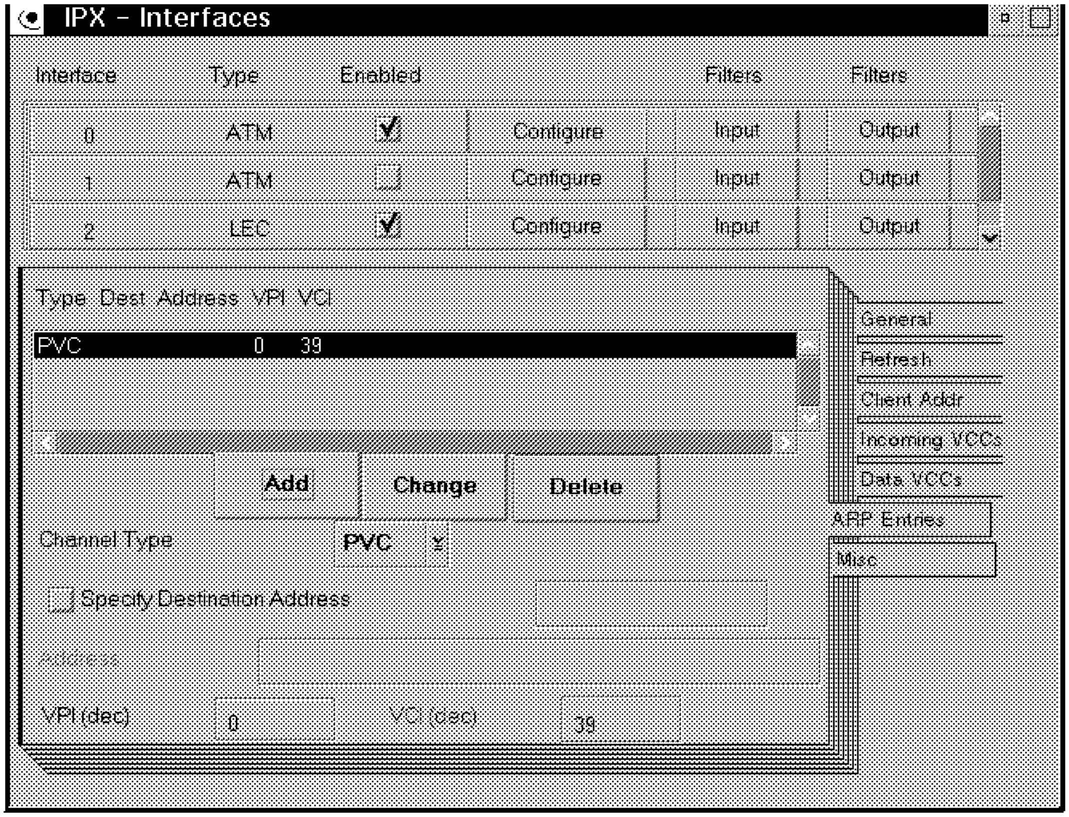
□ Use Runtime Generated Selector

Assigned Selector (hex) A1

Figure 149. Configure RFC 1483 IPX Client

Configuring an ATMARP IPX client is required when you use VCCs to connect to other (RFC 1483 compliant) IPX routers. Per ATM port you can define a single ATMARP IPX client. To simplify the definition of ATM addresses on remote routers, we use our user-defined ESI, disabled the use of a runtime generated selector, and set the selector byte to A1. Note that the ESI is also used as a MAC address to comprise the IPX address for this interface (see 8.1, “IPX Addresses” on page 205).

Next step is to define the PVC and SVC to the remote routers. For this purpose we selected ARP Entries. Figure 150 on page 230 appeared.



IPX - Interfaces

Interface	Type	Enabled		Filters	Filters
0	ATM	☑	Configure	Input	Output
1	ATM	□	Configure	Input	Output
2	LEC	☑	Configure	Input	Output

Type	Dest Address	VPI	VCI
PVC		0	39

Channel Type: PVC

□ Specify Destination Address

Address

VPI (dec): 0

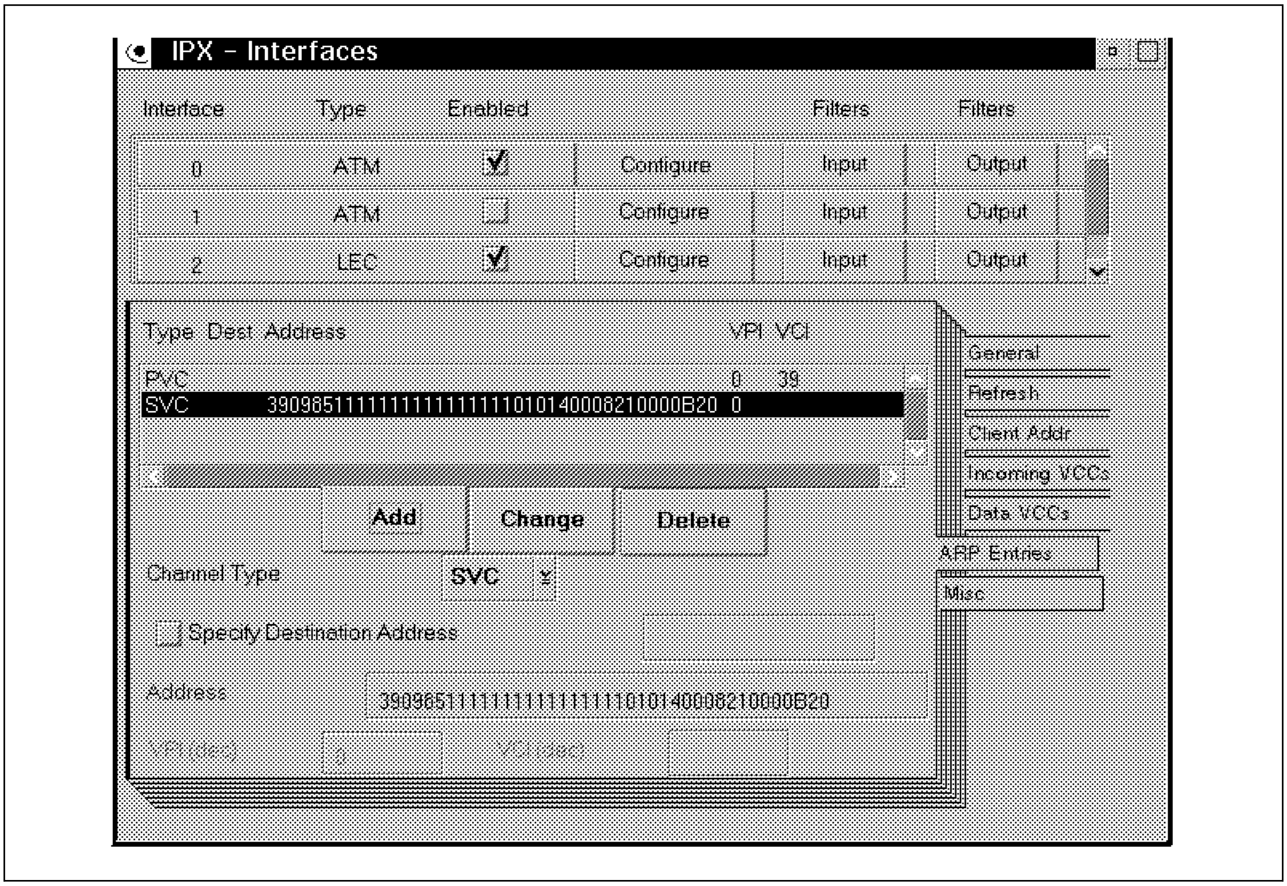
VCI (dec): 39

Figure 150. Configure PVC

A PVC entry was added by entering the appropiate VPI and VCI number. Make sure they match the definitions on your adjacent ATM switch. We disabled Specify Destination Address. Instead InATMARP will be used to learn the IPX address of the PVC-connected router. Selecting Add makes the definitions permanent.

Note: Multiple RFC 1483 IPX PVCs can be defined per ATM port.

Last step is to define the SVC entry. For this purpose we set the Channel Type to SVC. Figure 151 on page 231 appeared.



IPX - Interfaces

Interface	Type	Enabled		Filters	Filters
0	ATM	☑	Configure	Input	Output
1	ATM	□	Configure	Input	Output
2	LEC	☑	Configure	Input	Output

Type	Dest Address	VPI	VCI
PVC		0	39
SVC	3909851111111111111111010140008210000B20	0	

Channel Type: SVC

□ Specify Destination Address

Address: 3909851111111111111111010140008210000B20

VPI (dec): 0

VCI (dec):

Figure 151. Configure SVC

A SVC entry was added by entering the ATM address of the remote ATMARP IPX client. We disabled Specify Destination Address. Instead InATMARP will be used to learn the IPX address of the SVC-connected router. Selecting Add makes the definitions permanent.

Note: Multiple RFC 1483 IPX SVCs can be defined per ATM port.

This concludes the PVC and SVC definitions and our IPX configuration. For the remaining screens we used default parameters:

- Refresh

  ```
  Refresh timeout: 5 minutes
  Enable auto-refresh
  ```

- Incoming VCCs

  ```
  Do not validate peak cell rate of best effort VCCs
  ```

- Data VCCs

  ```
  Traffic type:  Best effort
  Peak cell rate: 155000 Kbps
  ```

- Miscellaneous

  ```
  Maximum SDU:  9188
  ```

Chapter 9. MSS Server and Bridging

This chapter briefly describes the main bridging methods, lists the bridging functionality provided by the MSS Server, and explains the commands available to customize the MSS Server′s bridging function. In addition, MAC address filtering will be covered, with a brief description of its operation and a list of available configuration commands.

For both subjects, a simple example is included to illustrate the configuration effort required using the Configuration Program. Configuration examples using the command line interface can be found in Chapter 11, "Implementation Scenarios" on page 289.

Further information on the MSS Server′s bridging functions can be found in the *Multiprotocol Switched Services (MSS) Server Command Line Interface Volume 2*.

9.1 Introduction to Bridging

Important

The MSS Server is capable of bridging between ELANs. As its bridging functions are essentially the same as used by convential LAN bridges, we give an introduction to legacy LAN bridging first.

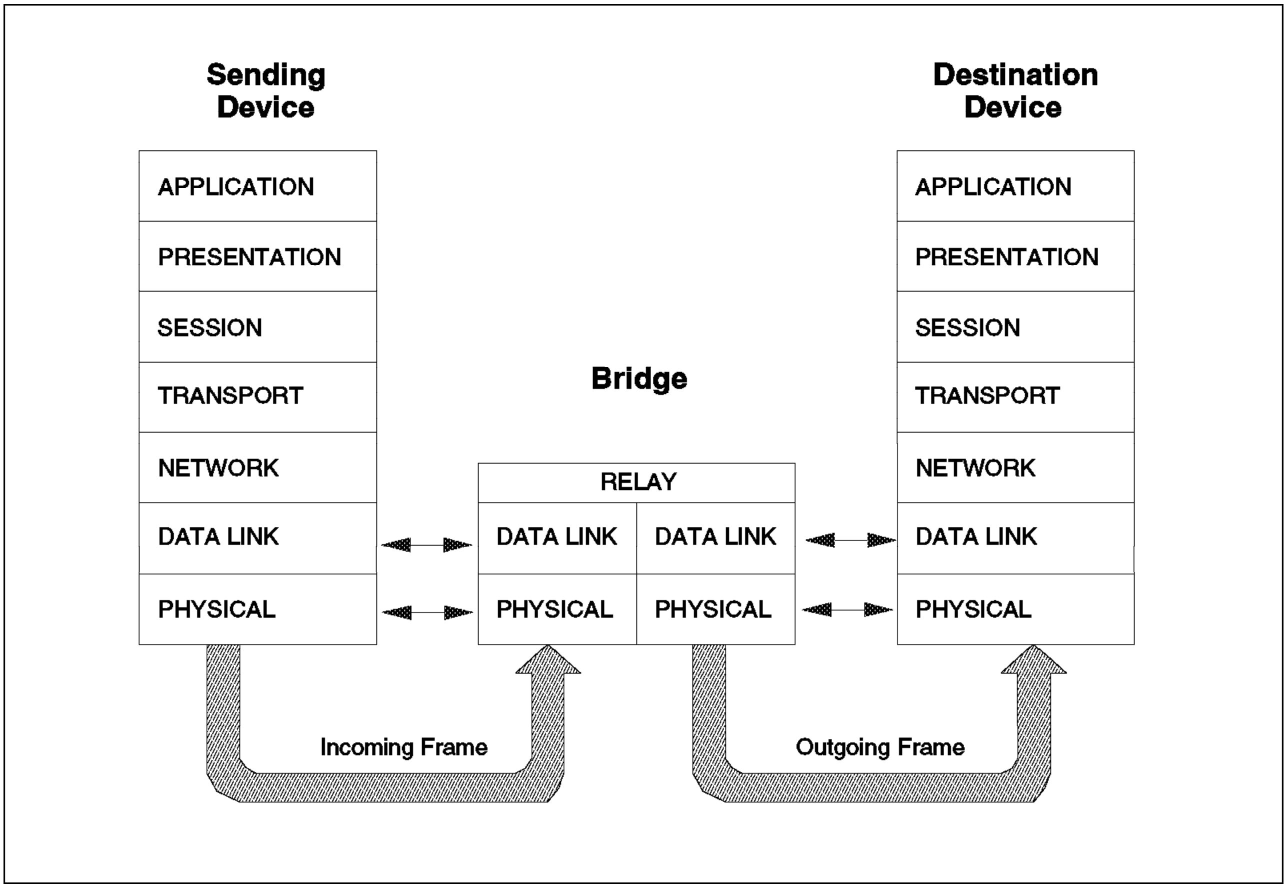


Figure 152. Bridge Implementation

Bridges act as data link layer relay between LANs. As shown in Figure 152, a bridge implements the physical and data link layers of the OSI reference model. A bridge participates as a device on the networks to which it is attached, exchanges information with devices on those networks and selectively forwards information between the networks.

There are the following specific disadvantages to bridging:

- Bridges offer no protection against large volumes of broadcast packets. In forwarding broadcast packets, bridges are only carrying out their normal function, but in doing so can impact internetwork performance and function. This can be a particular problem with remote bridges where broadcasts have to traverse inter-bridge serial links.
- Bridges have to drop packets that are too large for their attached networks. Bridges do not have the capability to fragment packets to accommodate networks with smaller maximum packet size.
- Bridges have no capability to provide congestion feedback to other bridges or to end nodes. This can lead to the need to discard packets with consequent impact on end system performance.

Equally there are specific advantages to bridging:

- Bridges are *plug and play* and require little expertise to install.
- Bridges require little administrative overhead. Once installed they generally function with minimum attention.
- Bridges are truly multiprotocol. They forward all packets and protocols irrespective of whether the protocols are routable or not.
- Bridges are generally transparent to end systems. Generally, no specific customization is required at the end system.

To summarize, bridges are best used to provide convenient local connection within single site networks, possibly where there is limited technical and administrative support for the products. They are particularly suitable for environments where it is undesirable or impractical to configure end systems for operation with routers, or where the protocols in use are mainly non-routable.

9.2 Bridging Methods

There are two primary methods of bridging:

- Transparent bridging (mainly used with Ethernet LANs) called also spanning tree bridging (STB)
- Source-route bridging (SRB) used in 802.5 LANs

From these two primary methods of bridging, there are other methods listed below:

- Source route transparent bridging (SRT)
- Source route translational bridging (SR-TB)
- Tunnel bridge (IP encapsulation)

All of these bridging methods are supported by the MSS Server. For more details about the bridging methods supported by the MSS Server please refer to *Multiprotocol Switched Services (MSS) Server Command Line Interface Volume 2*.

In the following sections, we provide a summary description of these bridging methods.

9.2.1 Transparent Bridging

A transparent bridge is also called a spanning tree bridge (STB).

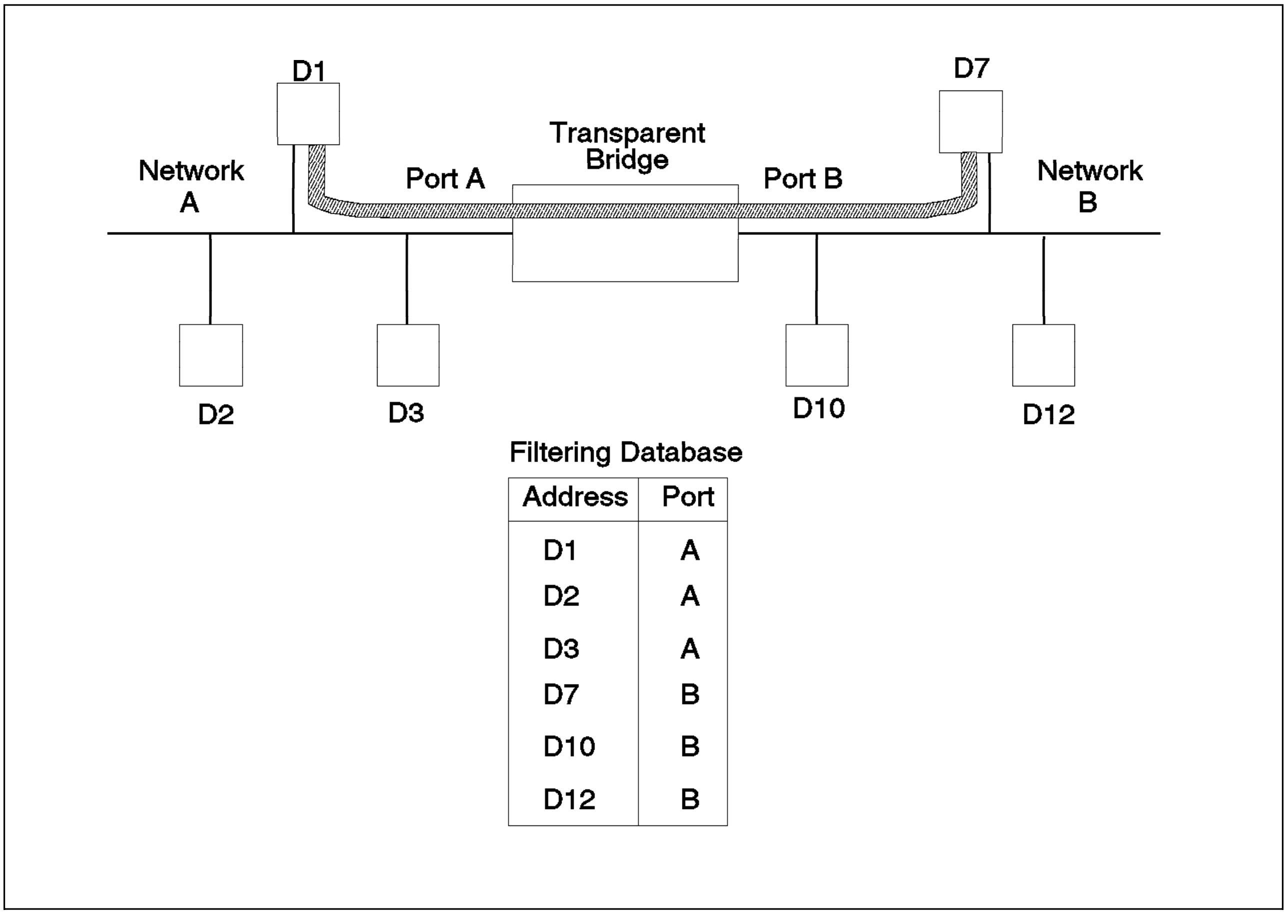
Transparent bridging is normally used to connect Ethernet LAN segments. It is specified in the ISO 8802-1 standard.

This form of bridging could also be used for connection of token-ring LAN segments although this is not common.

Transparent bridging is based on the principle that a sending device can transmit a frame to a receiving device on a LAN network without having any knowledge of the location of, or the path to, that receiving device.

Transparent bridges within a network are responsible for forwarding the frame to the correct destination by making the determination of whether a frame should be forwarded based on MAC sublayer destination address.

Transparent bridges achieve this by building and maintaining a *filtering database* that acts as a *forwarding table* for received frames. They build their database by copying all frames from the LANs to which they are attached and learning the location of devices by inspecting the MAC sublayer *source address* in each received frame.



Filtering Database

Address	Port
D1	A
D2	A
D3	A
D7	B
D10	B
D12	B

Figure 153. Transparent Bridging

Figure 153 illustrates how a transparent bridge will build up its filtering database. When the bridge receives a frame from device D1 on port A, it learns that D1 can be reached via the LAN on port A. Similarly, if a frame arrives from device D7 on port B, it learns that D7 can be reached via the LAN on port B.

For each new source address the bridge sees on the LANs, it adds an additional entry in its database. In time a full picture is built up of all devices on the two LANs and via which port they can be reached.

The bridge uses its filtering database to determine if an incoming frame should be forwarded or discarded. This is done by examining the MAC sublayer *destination address* of each frame and comparing it to the list of addresses in the filtering database:

- If the destination address is not in the database, the frame is forwarded on each port except the receiving port.
- If the destination address is in the database and the frame was received on a port associated with the address, the frame is discarded.
- If the destination address is in the database and the frame was received on a port not associated with the address, the frame is forwarded to the associated port for this destination address in the database.

Transparent bridges require that there be only a *single* active path between any two LANs in an internetwork. This requirement is to ensure that frames do not loop such that they are seen on both ports of a bridge. If this happens the bridge will be unable to forward the frames correctly to their destination.

Transparent bridges support and use the spanning tree protocol, which ensures a loop-free topology between all the transparent bridges within the network.

9.2.2 Source-Route Bridging (SRB)

Source-route bridging is implemented by IBM and compatible bridge products for use over token-ring LAN segments.

Source routing requires a sending device to specify the path that should be taken by a frame across an internetwork, rather than allowing the decision to be made by individual bridges. To do this a sending device must determine the best path to a destination and include it in all frames to that destination. The best path to a destination is found using a discovery process, one implementation of which is described below.

A sending device sends a discovery frame to the intended destination device marked single-route broadcast. Bridges in a token-ring internetwork should be configured using the token-ring spanning tree algorithm to permit only one path for single-route broadcast frames between devices. The destination device should therefore receive only a single copy of the discovery frame.

The destination device responds to the discovery frame with a discovery response frame marked all-routes broadcast. This will contain the most significant bit (the route information indicator, also called RII) set in the source MAC address field, and an entry in the routing information field (RIF). This will initially contain zero in the bridge number field, and the number of the networks to which the destination device is attached in the segment number field.

The discovery response frame, because it is marked all-routes broadcast, will pass through all bridges on its way back to the original sending device. Each bridge that the frame passes through must insert its bridge number and LAN segment, and hence the frames that return to the original sending device contain the routes they have taken through the bridged internetwork.

The routing information field can hold data about thirteen bridges and fourteen LAN segments. If a frame is received by a bridge with this field full it will be discarded. This limits the number of bridge *hops* in the network to thirteen, and consequently the maximum size of source-route bridged internetworks.

Figure 154 shows how the routing information field is used to define a route through an internetwork between end nodes D1 and D7.

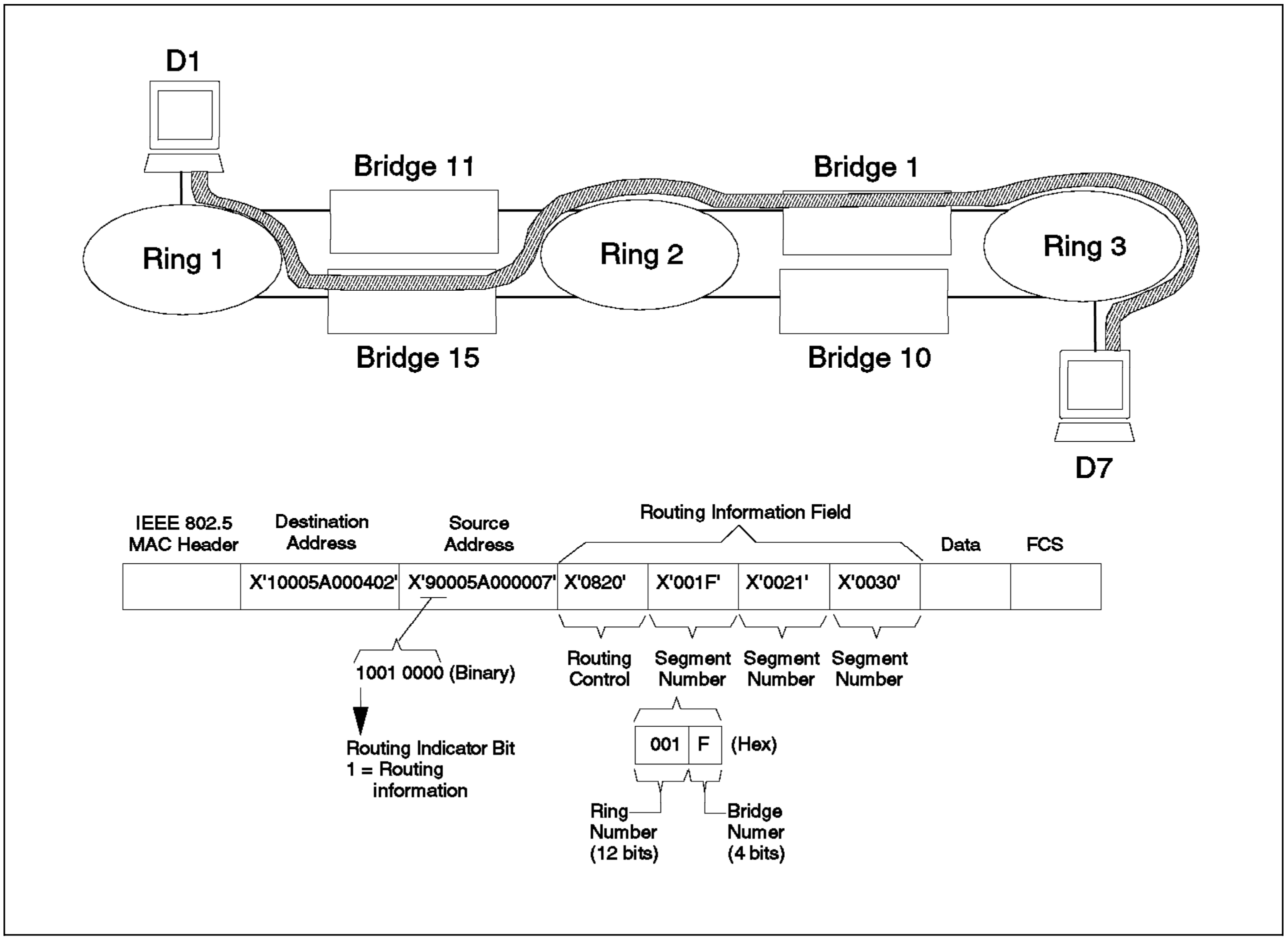


Figure 154. Source-Route Bridging

Note: The MSS Server supports the standard 802.5 13-hop maximum, but the bridge can be configured for any maximum hop count up to thirteen.

The original sending device therefore receives one or more discovery response frames. These frames contain routing control and bridge and LAN segment numbers in their routing information fields. The routing control field indicates the number of bridge/LAN segments in the routing information field, and also the maximum frame size that can be supported by the route.

The sending device can now select the best route to use through the internetwork to reach the destination device. Current implementations select the route in the *first* received discovery response frame (the fastest path at the time of the discovery process), although the architecture allows route selection based on other criteria, for example, maximum frame size supported by the route.

9.2.3 Source Route Transparent Bridging (SRT)

The IEEE 802.1 committee identified the need for source route bridges to interoperate with transparent bridges in the same internetwork.

A *source route transparent bridge (SRT)* standard has been defined to achieve this goal.

The principle behind SRT bridges is very simple. An SRT bridge inspects all received frames and looks for the presence of the routing information indicator (RII) and the routing information field (RIF). If these fields are present the SRT bridge uses them and acts as a source-route bridge. If not, the SRT bridge operates in transparent bridge mode and forwards frames based on their MAC sublayer destination address and its associated entry in the filtering database.

The source route transparent bridge does not allow source route bridge devices to communicate with transparent bridge devices. SRT bridge is the capability for its interfaces to understand both source-route bridging and transparent bridging devices. But an SRT bridge will never translate a source-route bridge frames into transparent bridge frames, and vice versa.

Figure 155 shows you how frames with RII=1 are forwarded with RII=1 and how frames with RII=0 are forwarded with RII=0.

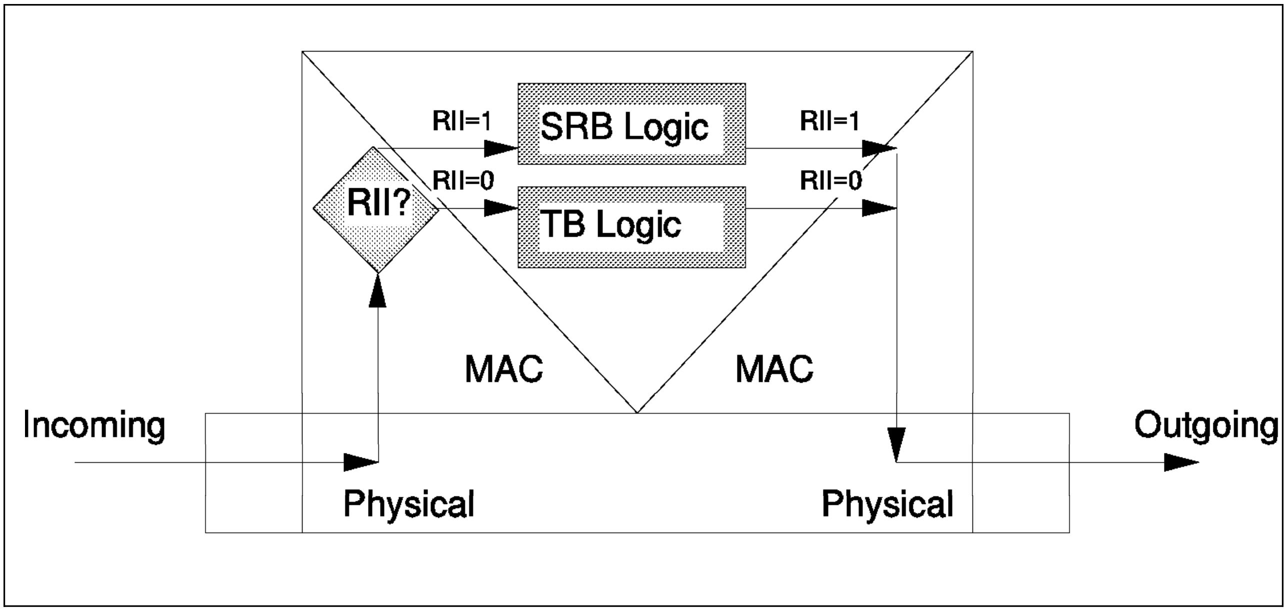


Figure 155. Source Route Transparent Bridge

9.2.4 Source Route Translational Bridge (SR-TB)

Source route translational bridge (SR-TB) is not an ISO standard definition. However, more and more bridges are implementing the SR-TB because of the need to interconnect source-route bridge domain with transparent bridge domain.

The goal of the source route translational bridge is to translate source-route bridge frame into a transparent bridge frame and vice versa.

The SR-TB bridges have to change the MAC layer protocol from (or to) Ethernet protocol to (or from) token-ring protocol. Actually, considering the ISO bridge definition, this translation does not belong to a bridge. But it has been implemented in a lot of bridges, in order to be able to interconnect a source-route bridge domain and a transparent bridge domain regardless of the protocol of the upper layer.

Figure 156 shows you that the SR-TB allows an SRB device with RII=1 to communicate with an STB device (RII=0).

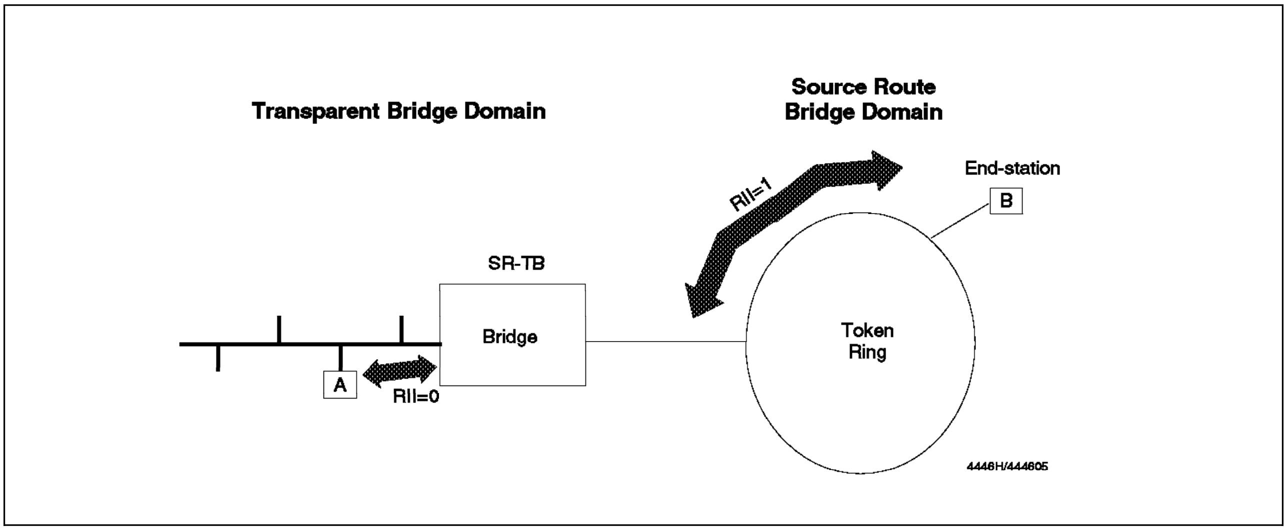


Figure 156. Source Route Translational Bridging

MSS

Translational bridging requires MAC address conversion (MSB to LSB vice versa) within control frames for specific protocols. IP and IPX implement this, NetBIOS and LLC do not. As the MSS Server does not support this MAC address conversion, SR-TB is limited to NetBIOS and LLC. LLC is, for example, used by SNA.

9.2.5 Tunnel Bridge

The tunnel bridge allows source-route bridge domains or transparent bridge domains to communicate across an IP network.

The tunnel bridge receives bridged frames from its source-route bridge or transparent bridge domain. The frames are encapsulated into IP datagrams that are sent to the destination IP address. These IP datagrams are routed in the IP network as any other IP datagrams with the IP rules.

The destination IP address is actually another bridge implementing the tunnel bridge feature. This target bridge removes the IP envelope from these IP datagrams making them source-route bridge or transparent bridge frames. Then the target bridge sends these frames to its source-route bridge domain or transparent bridge domain as any other bridged frames.

Figure 157 shows you an example of the tunnel bridge implementation.

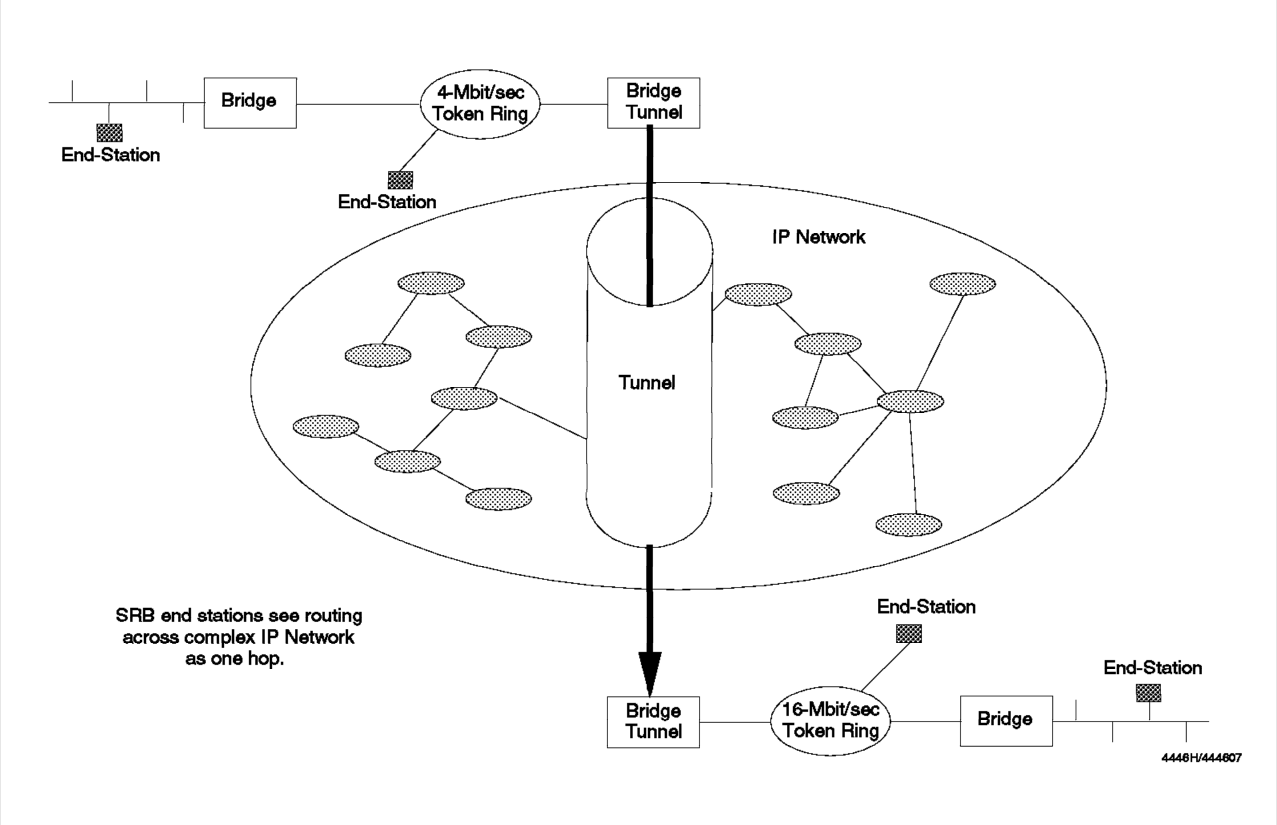


Figure 157. Tunnel Bridging

With tunnel bridging, as far as the source-route bridge is concerned, the IP network is seen as a single LAN segment, regardless of the complexity of the IP network. Then it adds only one hop to cross this IP network.

The number of hops from the source device to the source IP tunnel bridge, plus one hop to cross the IP network, plus the number of hops from the destination IP tunnel bridge to the destination device, must not exceed the thirteen hops count limitation of the source-route bridge implementation.

9.3 Bridging with the MSS Server

Bridging requires an emulated LAN environment as Classical IP only supports IP routing.

Note: A classical IP connection between two MSS Servers can be used to *tunnel* bridging traffic. For details about tunneling see 9.2.5, “Tunnel Bridge” on page 239.

The MSS Server allows you to configure the bridging methods supported by means of the Adaptive Source Routing Transparent (ASRT) bridge.

The ASRT bridge is a software collection of several bridging options. It produces a *bridge personality*, also called bridge behavior, through the configuration of parameters for the bridge and all its interfaces.

Figure 158 shows an example of how to list the bridge behavior using the MSS command line interface.

```
*
*talk 6

Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>list bridge

                Source Routing Transparent Bridge Configuration
                ===============================================

Bridge:                 Enabled                  Bridge Behavior: SRT      1
                   +----------------------------+
-------------------| SOURCE ROUTING INFORMATION |-------------------------------

                   +----------------------------+
Bridge Number:           01                      Segments:           1
Max ARE Hop Cnt:         14                      Max STE Hop cnt:    14
1:N SRB:                 Not Active              Internal Segment:   0x000
LF-bit interpret:        Extended
                   +-------------------+
-------------------| SR-TB INFORMATION |----------------------------------------

                   +-------------------+
SR-TB Conversion:        Disabled
TB-Virtual Segment:      0x000                   MTU of TB-Domain:   0
                   +-------------------------------------+
-------------------| SPANNING TREE PROTOCOL INFORMATION |-----------------------

                   +-------------------------------------+
Bridge Address:          Default                 Bridge Priority:    32768/0x8000

STP Participation:       IEEE802.1d on all ports
                   +-------------------------+
-------------------| TRANSLATION INFORMATION |----------------------------------

                   +-------------------------+
FA<=>GA Conversion:      Enabled                 UB-Encapsulation:   Disabled
                   +------------------+
-------------------| PORT INFORMATION |-----------------------------------------

                   +------------------+
Number of ports added: 2
Port:   1      Interface:       1      Behavior:   STB & SRB        STP:  Enabled   2
Port:   2      Interface:       4      Behavior:    STB Only        STP:  Enabled   2

ASRT config>
```

Figure 158. List Bridge Configuration

Note:

1 This is the bridge personality.

2 This is the port level bridge configuration.

The bridging personality may have the following values:

- SRB (pure source routing bridge)
- STB (pure transparent bridge)
- SRT (source route transparent bridge)
- SR-TB (source route translational bridge)
- ASRT (either SRB, STB, SRT or SR-TB, depending on the source and the destination devices)
- Unknown (when the bridge customization results in no valid bridging)

In the MSS Server, a bridge port is associated with a logical interface. This interface is created when defining a LEC on the MSS Server. Therefore, before specifying the bridge port parameters you must configure the LE client it is running atop. Note that the LE client can be either a token-ring or an Ethernet LEC, and the corresponding bridge port is, therefore, a token-ring or an Ethernet bridge port, respectively.

If you have your LECs configured, you can enable bridging and configure the LEC to perform one of the bridging methods.

Table 10 lists the configuration settings to define the desired bridge personality on the logical interfaces. In this table, bridge ports A, B and C can be Ethernet or token-ring LE clients.

Note that the only bridging method supported by the Ethernet emulated LAN interface is STB. A token-ring emulated LAN interface supports STB, SRB or SRT.

Table 10 (Page 1 of 2). Bridge Personality Regarding Bridge Configuration

Port A	**Port B**	**Port C**	**SR <-> TB Conversion**	**Bridge personality**
STB	STB	No bridging	Disable	STB
STB	SRB	No bridging	Enable	SR-TB
STB	STB & SRB	No bridging	Enable	ASRT
SRB	SRB	No bridging	Disable	SRB
SRB	STB & SRB	No bridging	Enable	ASRT
STB & SRB	STB & SRB	No bridging	Disable	SRT
STB & SRB	STB & SRB	No bridging	Enable	ASRT
STB	STB	STB	Disable	STB
STB	STB	SRB	Enable	SR-TB
STB	STB	STB & SRB	Enable	ASRT
STB	SRB	SRB	Enable	SR-TB
STB	SRB	STB & SRB	Disable	SRB
STB	SRB	STB & SRB	Enable	ASRT
STB	STB & SRB	STB & SRB	Disable	SRB

Table 10 (Page 2 of 2). Bridge Personality Regarding Bridge Configuration

Port A	**Port B**	**Port C**	**SR <-> TB Conversion**	**Bridge personality**
STB	STB & SRB	STB & SRB	Enable	ASRT
SRB	SRB	SRB	Disable	SRB
SRB	SRB	STB & SRB	Enable	ASRT
SRB	STB & SRB	STB & SRB	Disable	SRT
SRB	STB & SRB	STB & SRB	Enable	ASRT
STB & SRB	STB & SRB	STB & SRB	Disable	SRT
STB & SRB	STB & SRB	STB & SRB	Enable	ASRT

When a LEC is acting as a bridge port, it joins the ELAN as proxy and registers its MAC address (regardless of its mode). If the bridging mode is SRB or SRT, a route descriptor is also registered with the LES. The route descriptor is comprised of a 12-bit LAN segment number (000-FFF) and a 4-bit bridge number (0-F).

9.4 Bridging - A Configuration Example

To illustrate how to configure a bridged environment using the Configuration Program, we define source routing between two emulated token-ring LANs. In our example, the LE clients connect directly to the LES without using an LECS. To simplify the definitions, we assume that the LES for both ELANs are external.

Figure 159 on page 244 shows this configuration as well as the most important parameters.

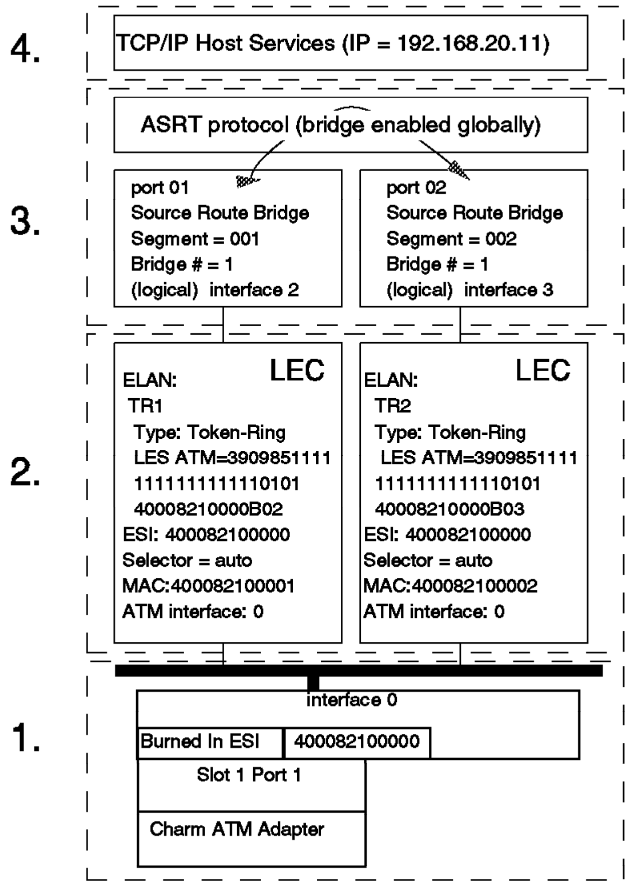


Figure 159. An Example of Source-Route Bridging

Note: In Chapter 11, "Implementation Scenarios" on page 289 you can find examples of bridging configurations using the command line interface.

Configuring bridge functions can be divided into four steps:

1. ATM interface configuration

   All traffic of each MSS Server′s bridge port is associated with a specific physical port. Configuration of each ATM port on which bridging functions are required is needed.

   Note: You can have multiple logical bridge ports on a single ATM port.

2. LEC configuration

   Bridging traffic will run on top of the LE client functions provided by the MSS Server. For each bridge port define a separate LE client.

3. Bridge configuration

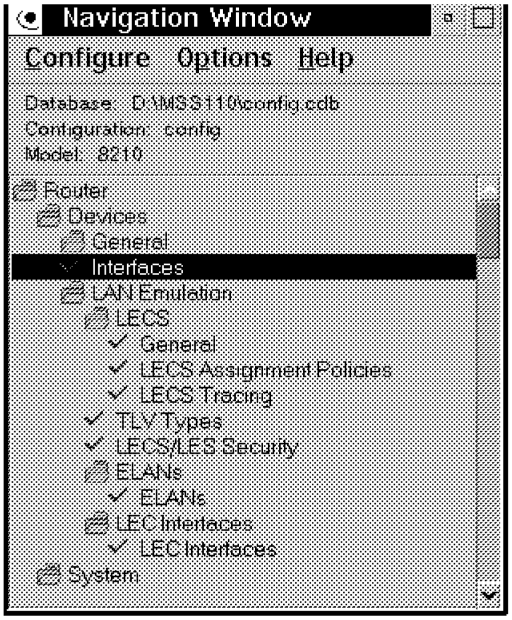
   The MSS Server is a multiport bridge. For each LE client interface defined, configure the associated bridge port parameters.

4. TCP/IP host services (optional)

   If no IP address has been assigned to the MSS Server and in-band IP connectivity is required, for example, to the management facilities, activate the TCP/IP host services.

9.4.1 ATM Port Definition

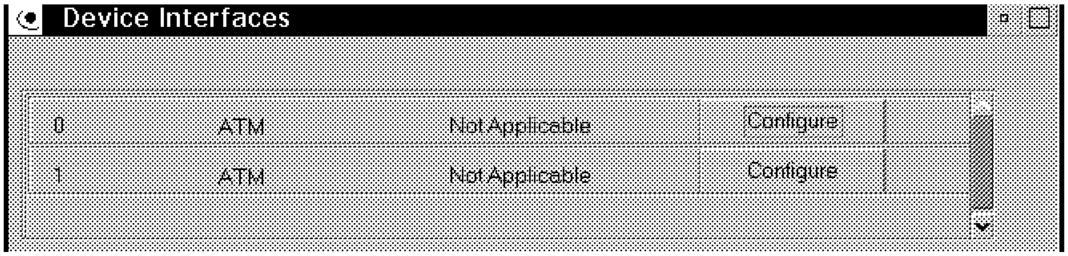
To start the ATM interface configuration select Interfaces in the General folder in the Navigation Windows as shown in Figure 160.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 160. Configure ATM Interface

Figure 161 will appear.

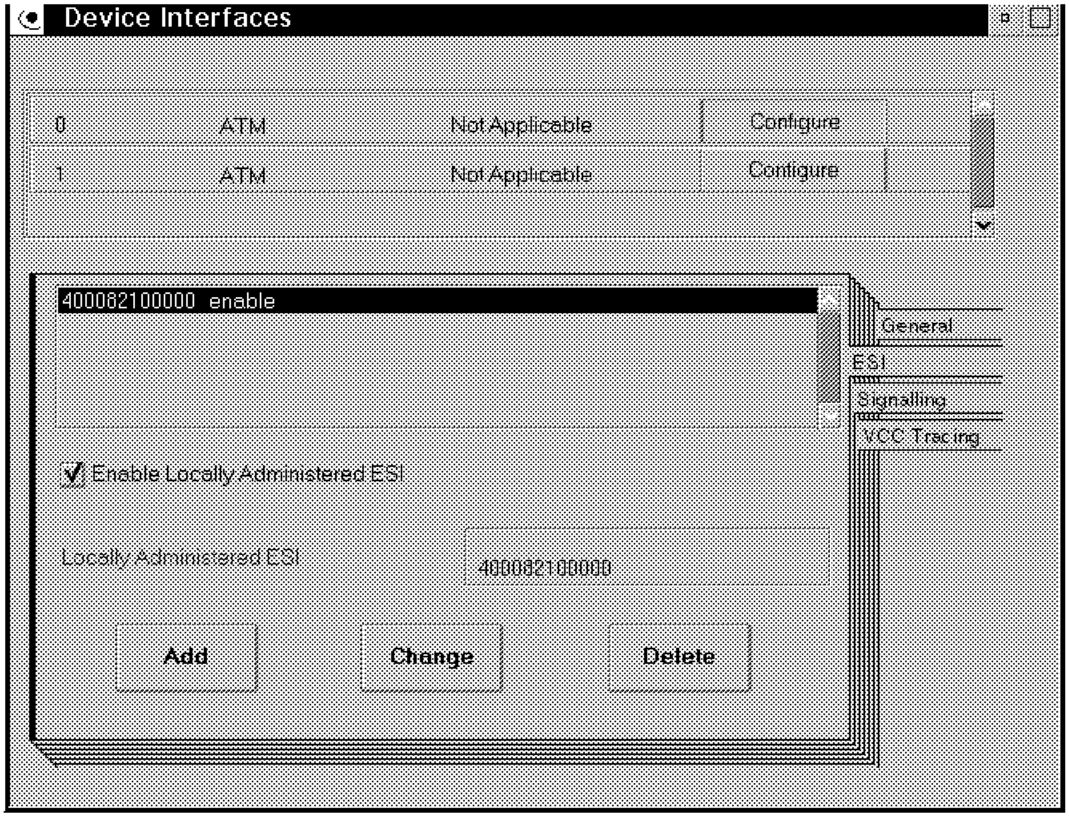


0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

Figure 161. ATM Interface

To simplify problem determination we decided to use a user-defined ESI for the LE clients used for bridging. If you want to use the *burned-in* ESI, you can skip this step.

To add an ESI click on the Configure button beside the interface you are planning to use (0 in our case). After selecting ESI, Figure 162 on page 246 appears.



0	ATM	Not Applicable	Configure
1	ATM	Not Applicable	Configure

400082100000 enable

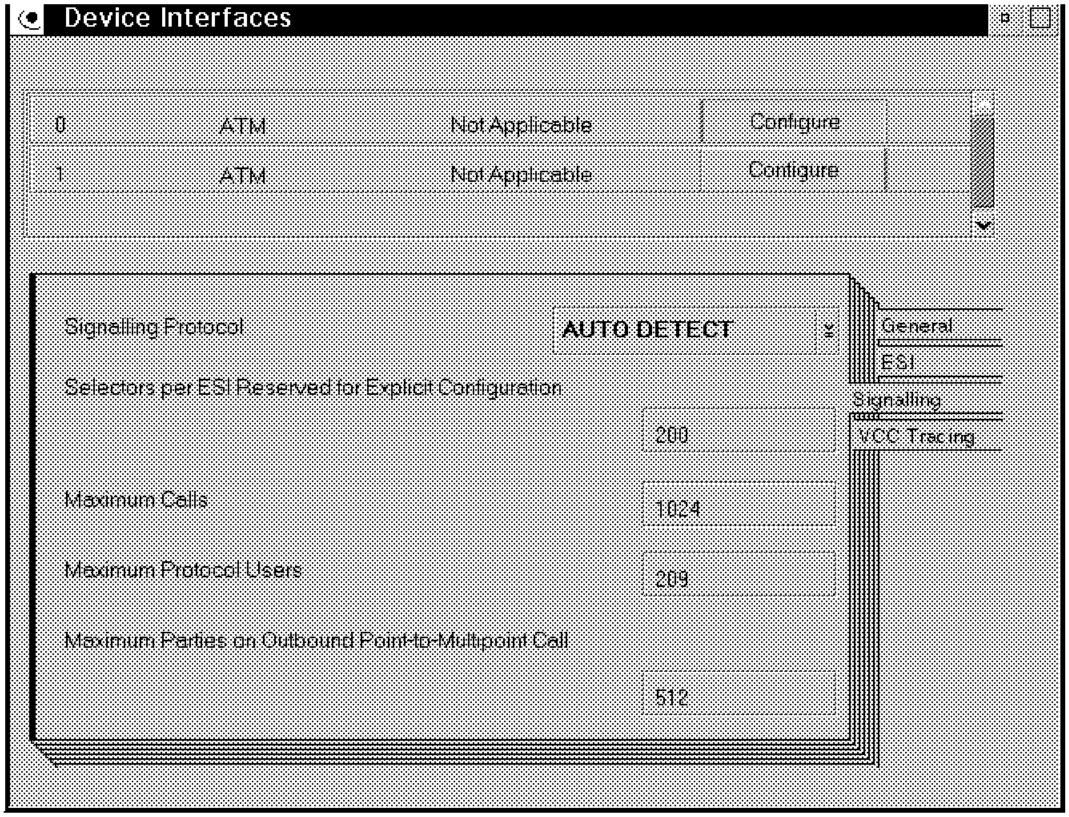
☑ Enable Locally Administered ESI

Locally Administered ESI 400082100000

Figure 162. Adding an ESI

Here you enter the locally-administrated ESI and then click on Add.

The next step is to configure the UNI version for this interface. After selecting Signalling, Figure 163 on page 247 appears.



0	ATM	Not Applicable
1	ATM	Not Applicable

Signalling Protocol: AUTO DETECT

Selectors per ESI Reserved for Explicit Configuration: 200

Maximum Calls: 1024

Maximum Protocol Users: 209

Maximum Parties on Outbound Point-to-Multipoint Call: 512

Figure 163. Configure the UNI Version

Select the appropriate UNI version (Signalling Protocol). If you are not sure about the UNI version that has been configured on the adjacent ATM switch port, select the auto detect option. This ends the configuration of the ATM interface.

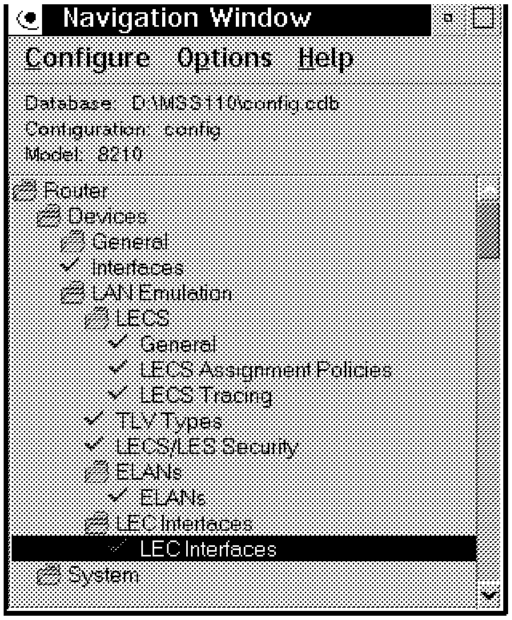
Note: If the UNI version is incorrect the MSS Server may not work (see Chapter 3, “MSS Server and ATM Ports” on page 53).

9.4.2 LE Client Definition

The next step is to configure the two LE clients that will be used as bridge ports.

Note: Because the definitions are almost identical, we detail the configuration of one token-ring LE client only.

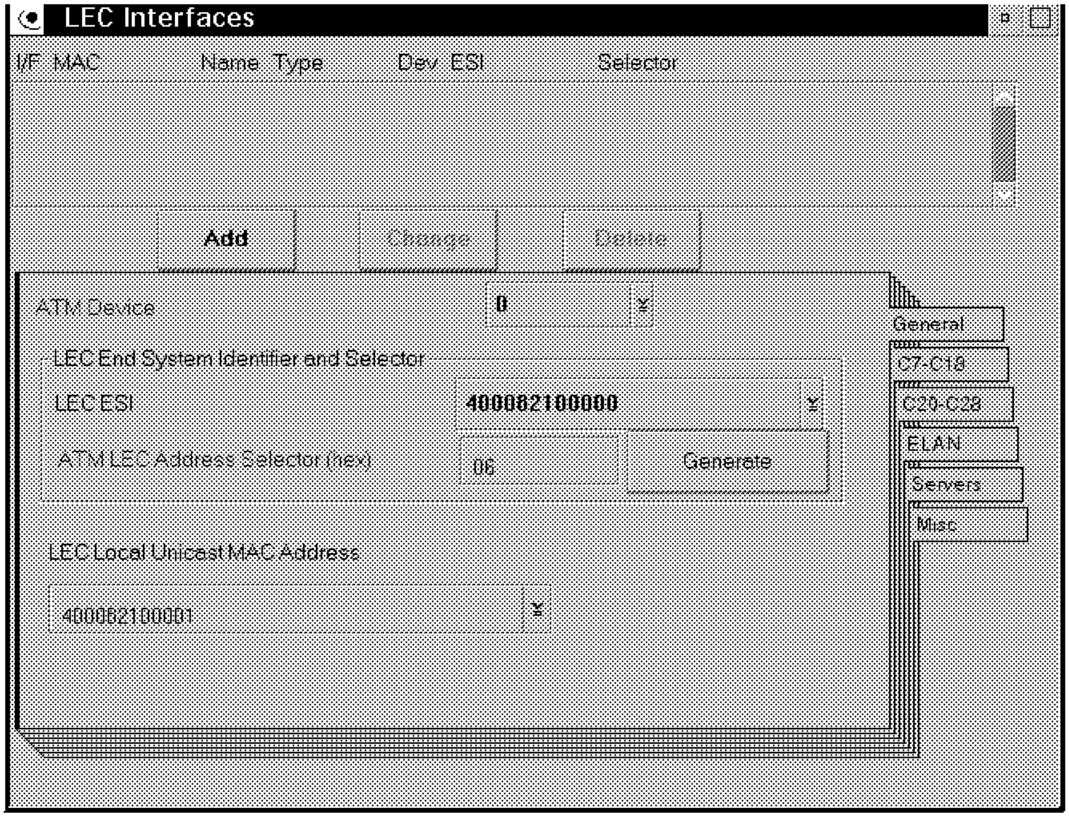
To start the LE client configuration select LEC Interfaces in the Navigation Window, as shown in Figure 164 on page 248.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 164. Configure LEC

As a result Figure 165 will appear.



ATM Device: 0

LEC End System Identifier and Selector

LEC ESI: 400082100000

ATM LEC Address Selector (hex): 06

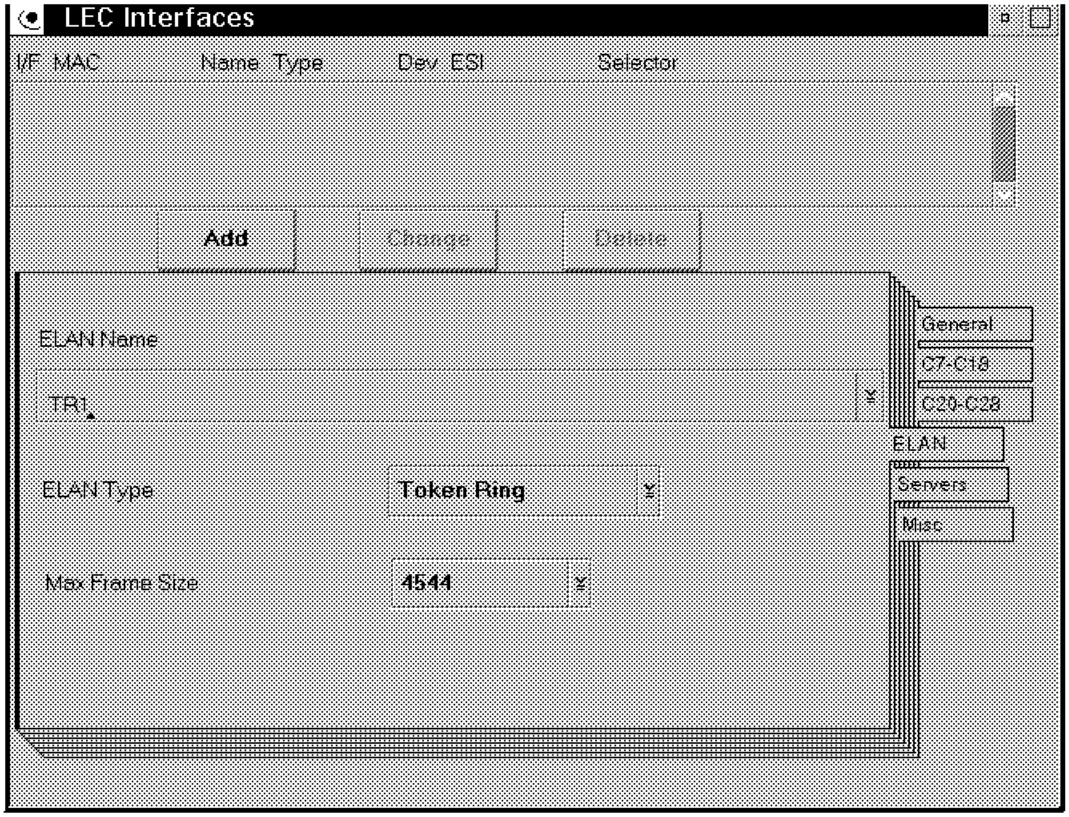
LEC Local Unicast MAC Address: 400082100001

Figure 165. Configure ESI, MAC, and Selector Byte

Here we select the ATM port (device) which will be used by the LE client, select our user-defined ESI, have the configurator generate a selector byte, and enter a MAC address. Note that the MAC address must be unique in your network.

Selecting ELAN results in Figure 166. This configuration screen enables us to enter the ELAN type, the maximum frame size, and the ELAN name. Because our LE client connects directly to the LES, the ELAN name is optional.

Note: For documentation purposes it is advised that you use the same name as defined on the LES and LECS.



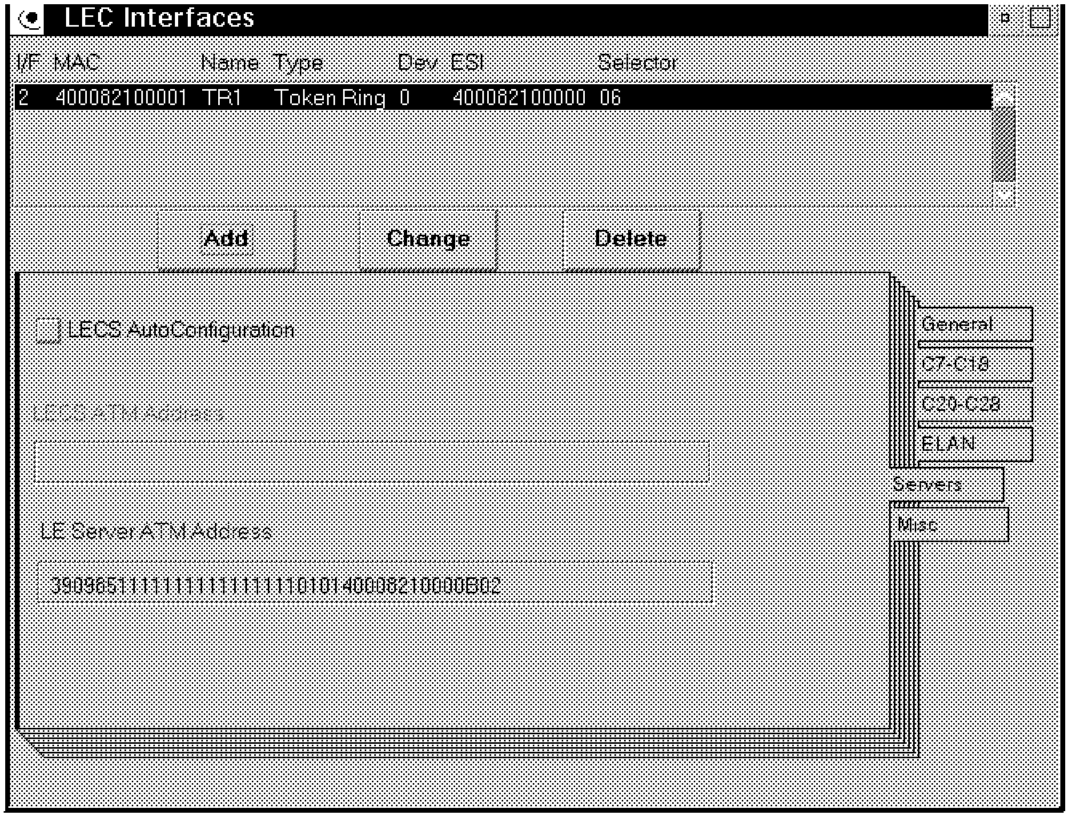
ELAN Name: TR1

ELAN Type: Token Ring

Max Frame Size: 4544

Figure 166. Configure ELAN Type, Name, and Maximum Frame Size

The next step is to configure the ATM address of the LES of the token-ring ELAN to which the LE client connects. To do so we select Servers. As a result Figure 167 on page 250 appears.



```
I/F  MAC           Name  Type        Dev  ESI           Selector
2    400082100001  TR1   Token Ring  0    400082100000  06
```

□ LECS AutoConfiguration

LECS ATM Address

LE Server ATM Address

3909851111111111111111010140008210000B02

Figure 167. Configure LES ATM Address

We configure the correct 20-byte LES ATM address and, because this LE client does not use LECS functions, disable LECS AutoConfiguration.

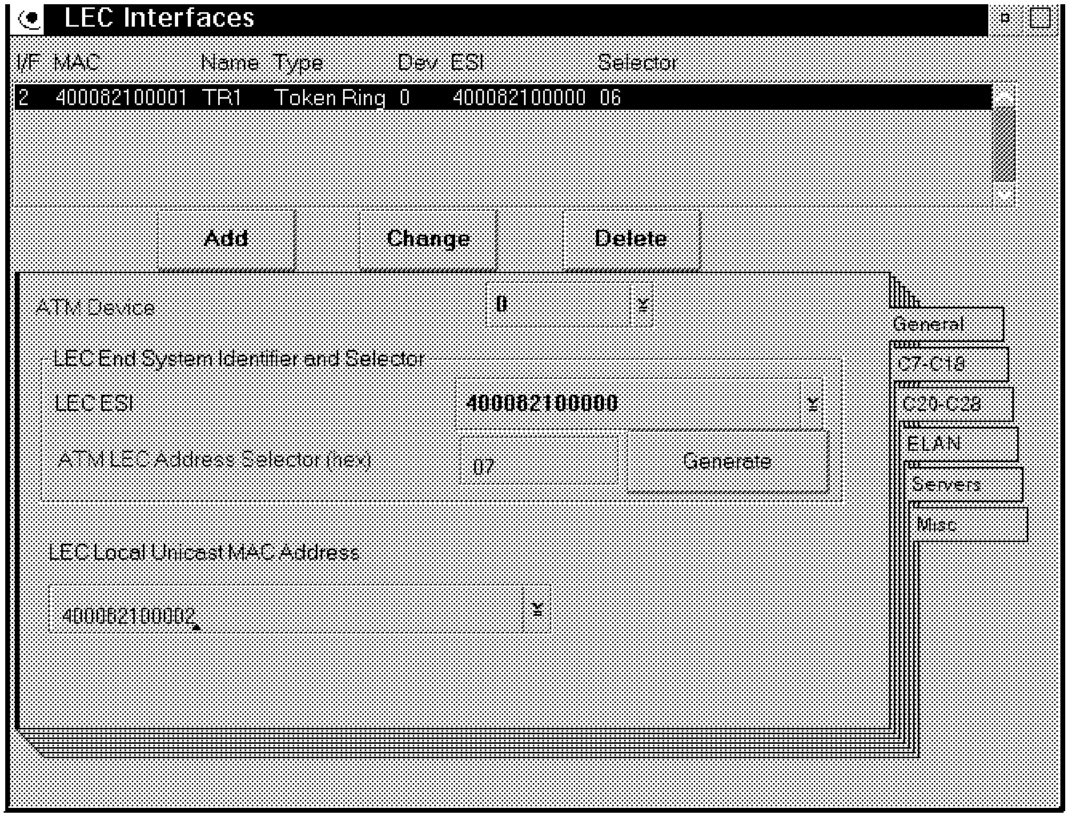
Selecting Add concludes the definition of the first token-ring LE client.

As indicated in the top line, a token-ring LE client has been added to which logical interface (I/F) 2 has been assigned.

I/F Number

Remember the logical interface number assigned during LE client definition, as it should be used during setting the bridge port parameters associated with this LE client.

The configuration of the second LE client can now be started. Figure 168 on page 251 appears after entering General.



LEC Interfaces

I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100001	TR1	Token Ring	0	400082100000	06

ATM Device: 0

LEC End System Identifier and Selector

LEC ESI: 400082100000

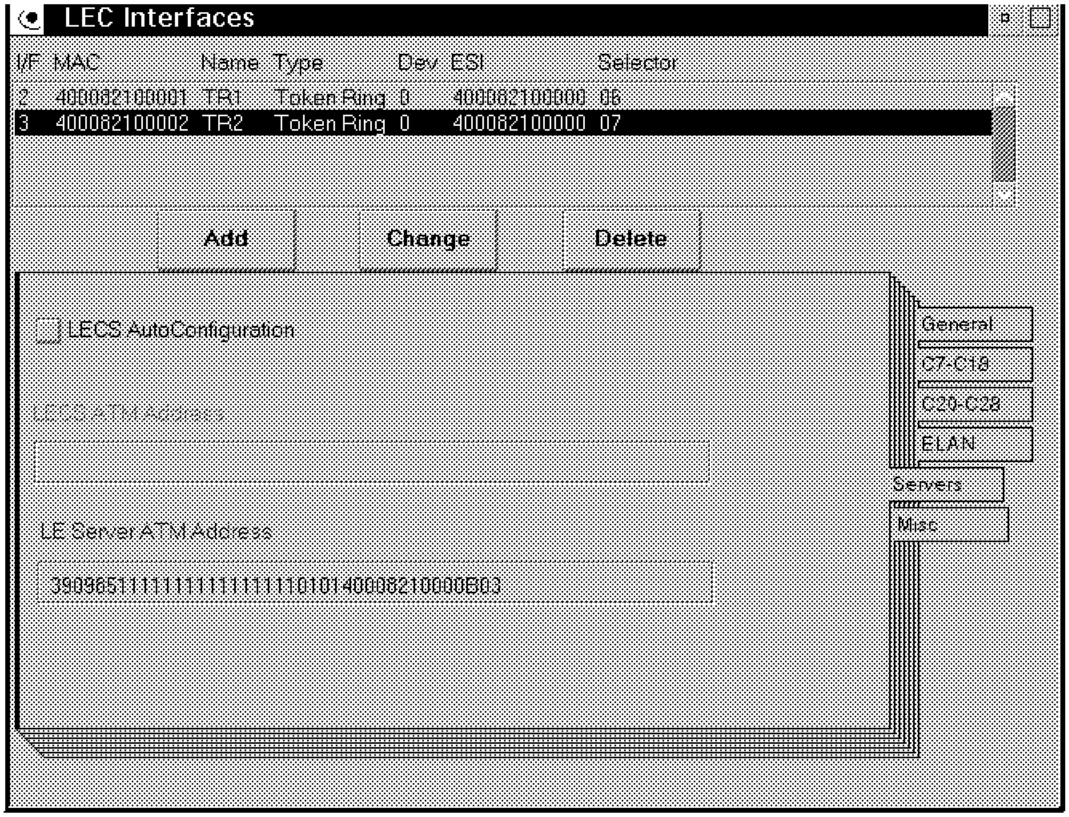
ATM LEC Address Selector (hex): 07

LEC Local Unicast MAC Address: 400082100002

Figure 168. Configure LE Client

We use the same ESI for both LE clients and, using the configurator to generate one, a selector that is one higher than that used by the first client. Make sure that the MAC address is unique.

The procedure to configure the second LE client is similar and, therefore, not repeated. The main differences are that in the definition of the second LEC interface you have to make sure that the LES ATM address enables the LE client to connect to the second ELAN (TR2 in our example).



I/F	MAC	Name	Type	Dev	ESI	Selector
2	400082100001	TR1	Token Ring	0	400082100000	06
3	400082100002	TR2	Token Ring	0	400082100000	07

□ LECS AutoConfiguration

LECS ATM Address

LE Server ATM Address

39098511111111111111110101 40008210000B03

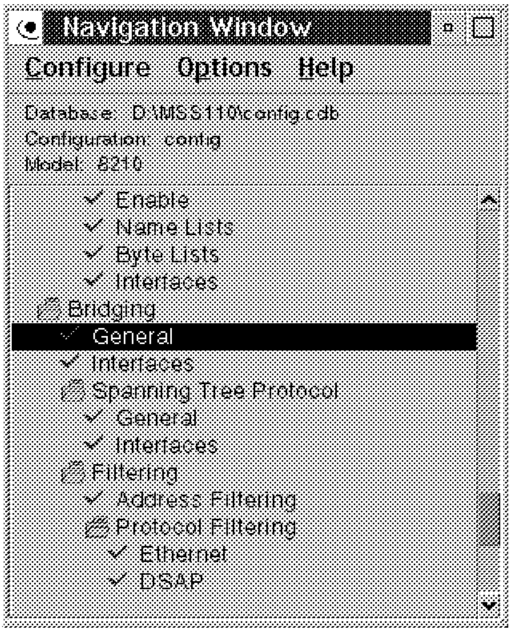
Figure 169. LES ATM Address Second LE Client

Figure 169 results after both LE clients have been configured. Note that the interface number (I/F) assigned to the second LE client is 3.

The next step is the configuration of the bridging protocol.

9.4.3 Bridging Configuration

To start the actual bridge configuration we select General in the Bridging folder on the Navigation Window, as shown in Figure 170 on page 253.

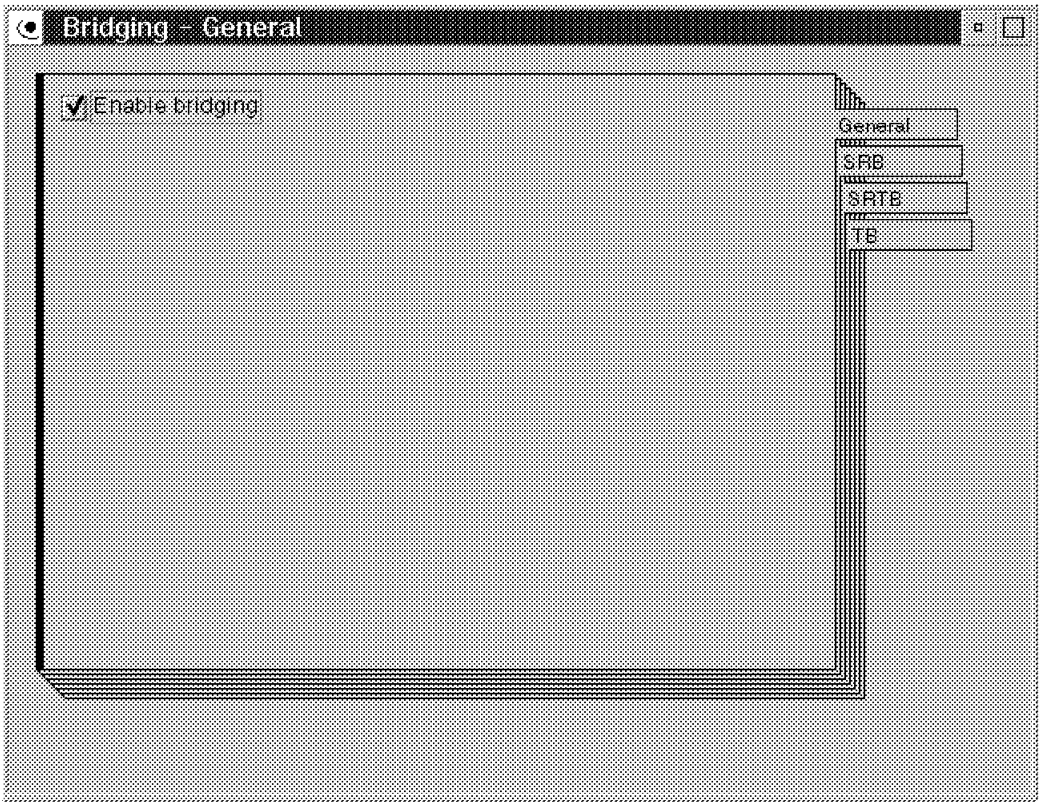


Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

```
        ✓ Enable
        ✓ Name Lists
        ✓ Byte Lists
        ✓ Interfaces
    Bridging
      ✓ General
      ✓ Interfaces
      Spanning Tree Protocol
        ✓ General
        ✓ Interfaces
      Filtering
        ✓ Address Filtering
        Protocol Filtering
          ✓ Ethernet
          ✓ DSAP
```

Figure 170. General Bridging Parameters

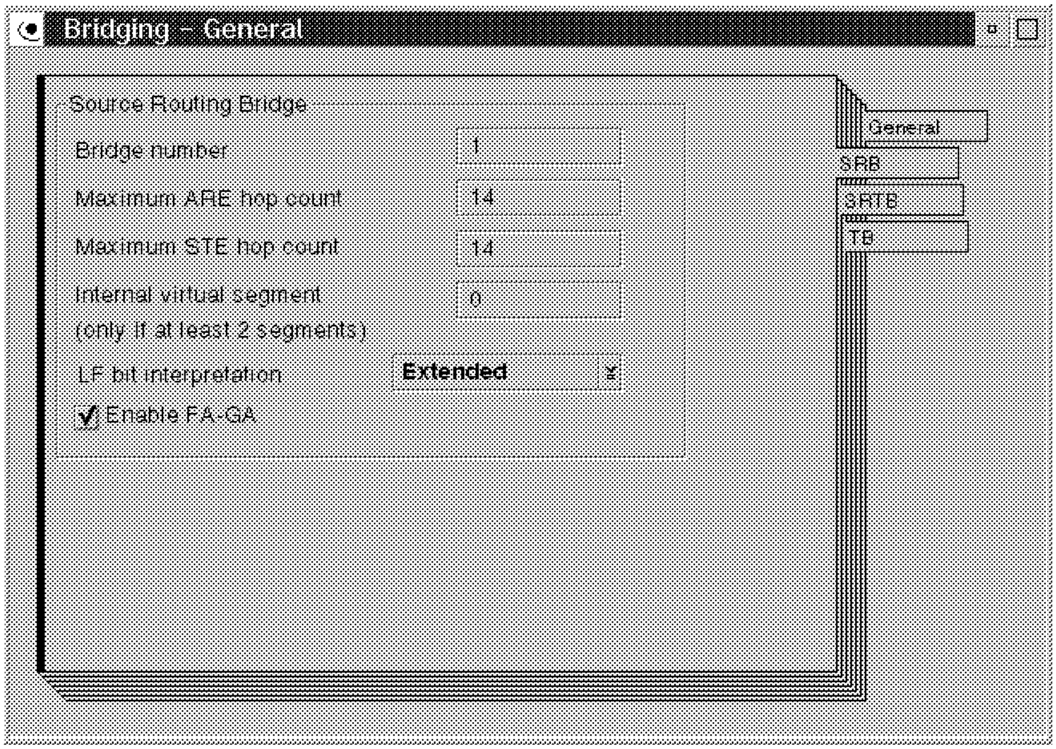
This brings you to the general bridging panel (see Figure 171).



☑ Enable bridging

Figure 171. Enable Bridge Globally

Select the Enable bridging option to enable bridge globally. General parameters for SRB, SRTB, and TB can be set to select the respective option on the right-hand side. After selecting SRB, Figure 172 on page 254 appears.



Bridging - General

Source Routing Bridge

Bridge number: 1

Maximum ARE hop count: 14

Maximum STE hop count: 14

Internal virtual segment (only if at least 2 segments): 0

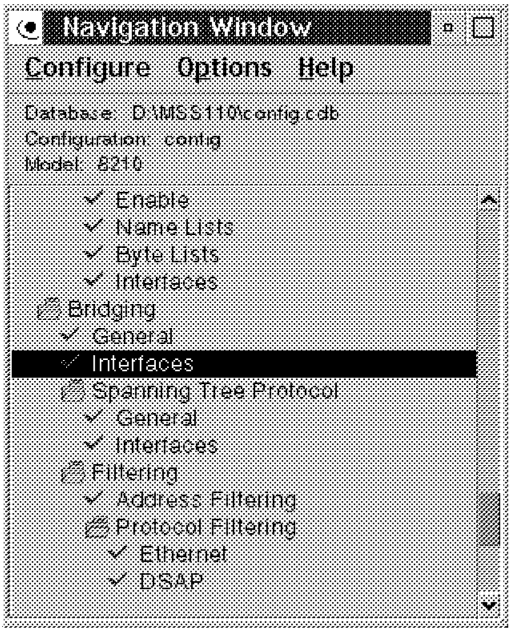
LF bit interpretation: Extended

☑ Enable FA-GA

Figure 172. Select Interfaces to Configure

Because in our example we are bridging between two bridge ports only, no internal virtual segment is required. We, therefore, set the Internal virtual segment parameter to zero. This concludes the definition of the SRB global parameters.

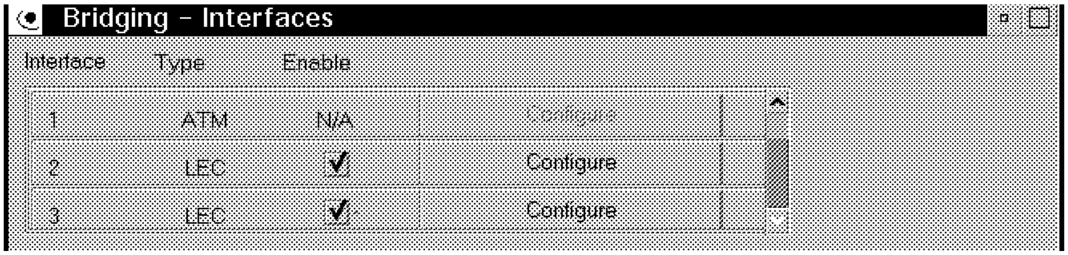
To set the port specific parameters, select Interfaces from the Bridging folder as depicted in Figure 173 on page 255.



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 173. Bridge Interfaces

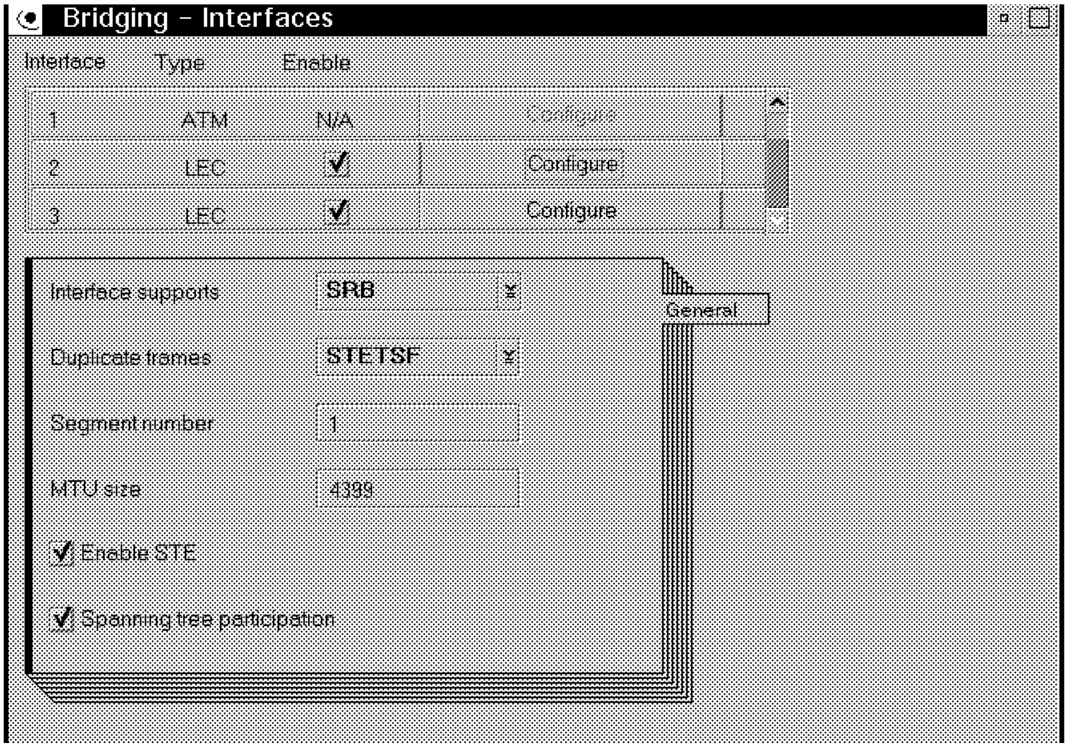
Figure 174 appears, showing all physical and logical interfaces configured. It includes the two token-ring logical interfaces defined during the previous LE client definition.



Interface	Type	Enable	
1	ATM	N/A	Configure
2	LEC	☑	Configure
3	LEC	☑	Configure

Figure 174. Physical and Logical Interface

After enabling a logical port for bridging, select Configure to define the bridge parameters for the specific interface. Figure 175 on page 256 appears when selecting logical interface 2, and Figure 176 on page 256 appears when selecting logical interface 3.



Bridging - Interfaces

Interface	Type	Enable	
1	ATM	N/A	Configure
2	LEC	☑	Configure
3	LEC	☑	Configure

Interface supports: SRB

Duplicate frames: STETSF

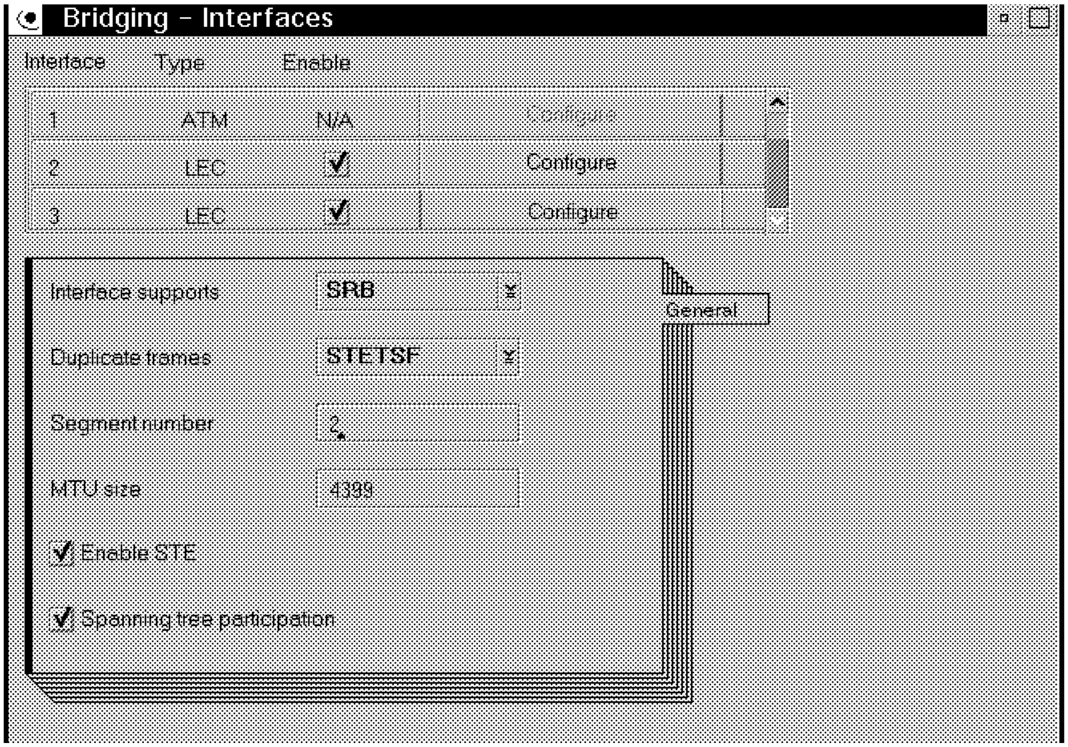
Segment number: 1

MTU size: 4399

☑ Enable STE

☑ Spanning tree participation

Figure 175. Bridge Parameters #1



Bridging - Interfaces

Interface	Type	Enable	
1	ATM	N/A	Configure
2	LEC	☑	Configure
3	LEC	☑	Configure

Interface supports: SRB

Duplicate frames: STETSF

Segment number: 2

MTU size: 4399

☑ Enable STE

☑ Spanning tree participation

Figure 176. Bridge Parameters #2

In each of these screens, specify the type of bridge and its related parameters. In our example we define source-route bridging (SRB). Make sure that the:

- LAN segment numbers are unique in your network (including your legacy LANs)
- Bridge numbers are unique per ELAN
- MTU is large enough to accommodate the largest frame sent on the ELAN

This concludes the bridge configuration.

9.4.4 TCP/IP Host Services

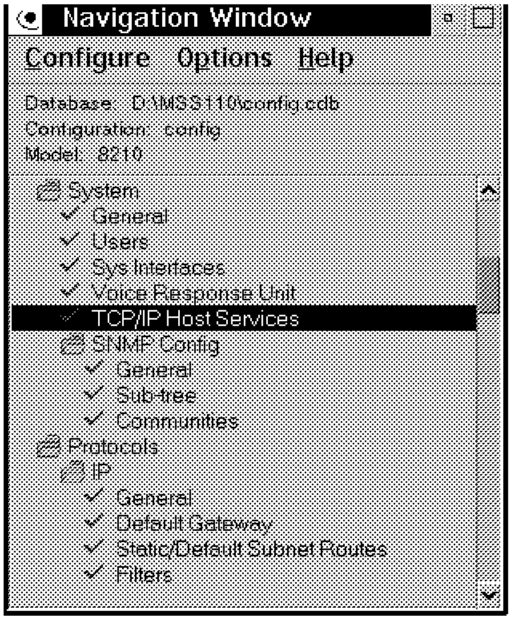
The activation of the TCP/IP host services enables you to assign an IP address to the MSS Server without conflicting with the bridging function enabled before.

Note: An alternative would be to add an IP address to an LE client that is also used for bridging. However, this disables the bridging function for IP.

Configuration of the TCP/IP host services enables IP connectivity via any of the bridging logical interfaces and enables in-band management services.

For information on TCP/IP host services, see *Multiprotocol Switched Services (MSS) Server Command Line Interface Vol 2*.

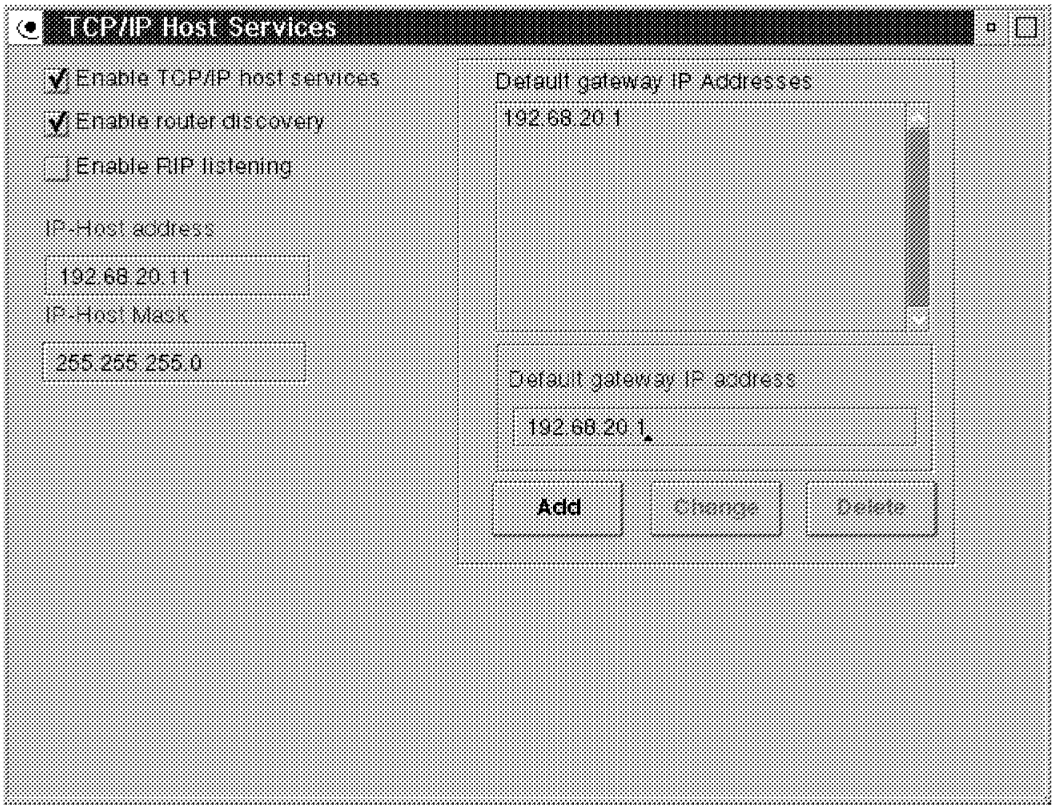
The configuration of the TCP/IP host services starts by selecting TCP/IP Host Services in the System folder within the Navigation Window (see Figure 177).



Database: D:\MSS110\config.cdb
Configuration: config
Model: 8210

Figure 177. Configure Host Services

Selecting this option results in Figure 178 on page 258.



☑ Enable TCP/IP host services
☑ Enable router discovery
□ Enable RIP listening

IP-Host address: 192.68.20.11
IP-Host Mask: 255.255.255.0

Default gateway IP Addresses: 192.68.20.1

Default gateway IP address: 192.68.20.1

Figure 178. Configure Host Services

Enter the IP address, subnet mask and the default gateway address for the MSS Server.

This ends the TCP/IP host services configuration.

9.5 Bridging Configuration Commands

The following sections summarize the bridging configuration commands and the tunnel bridge configuration commands available via the Config process of the MSS Server and the Configuration Program.

Table 11 shows you the main ASRT commands available. The *Multiprotocol Switched Services (MSS) Server Command Line Interface Volume 1* document covers these commands in detail.

Table 11 (Page 1 of 2). ASRT Configuration Commands

Command	Function
? (Help)	Lists all the ASRT configuration commands, or lists the options associated with specific commands.
Add	Adds station address entries to the permanent database, specific address mapping, ports, protocol filters, and a tunnel between endstations across an IP internetwork.
Change	Allows the user to change bridge and segment number.

Table 11 (Page 2 of 2). ASRT Configuration Commands

Command	Function
Delete	Delete station address entries, specific address mapping, LAN/WAN ports, protocol filters, and a tunnel between endstations across an IP internetwork.
Disable	Disables bridging functionality, duplicate frames, mapping between group and functional addresses, propagation of Spanning Tree Explorer Frames, source routing on a given port, reception of spanning tree explorer frames over a tunnel, conversion of source routed frame to transparent frame, transparent (spanning tree) bridging functionality on a given port, and a tunnel between bridges.
Enable	Enables bridging functionality, duplicate frames, mapping between group and functional addresses, propagation of Spanning Tree Explorer Frames, source routing on a given port, reception of spanning tree explorer frames over a tunnel, conversion of source routed frame to transparent frame, transparent (spanning tree) bridging functionality on a given port, and a tunnel between bridges.
List	Displays information about the complete bridge configuration or about selected configuration parameters.
NetBIOS	Allows access to the NetBIOS configuration prompt. See Table 13 on page 260 for details on the NetBIOS commands.
Set	Sets aging time for dynamic address entries, bridge address, maximum frame size for tunneling, largest frame (LF) bit encoding, maximum frame size, spanning tree protocol bridge and port parameters, Route Descriptor (RD) values, and filtering database size.
Tunnel	Allows access to the tunnel configuration prompt so that tunnel configuration commands can be entered. See Table 12 on page 259 about the tunnel bridge commands.
Exit	Exits the ASRT configuration process and returns to the CONFIG environment.

9.5.1 Configuration Commands for the Tunnel Bridging

Table 12 shows the list of the commands that you may use to configure tunnel bridging for the MSS Server.

Table 12 (Page 1 of 2). Tunnel Configuration Commands

Command	Function
? (Help)	Lists all the tunnel configuration commands, or lists the options associated with specific commands.
Add	Adds the unicast IP address of destination bridges participating in an IP unicast addressing configuration for bridging over IP.
Delete	Deletes the unicast IP address of destination bridges participating in an IP unicast addressing configuration for bridging over IP.
Join	Allows the MSS Server to join a peer-group, a client-group or a server-group to participate in an IP multicast addressing configuration for bridging over IP.
Leave	Allows the MSS Server to leave a peer-group, a client-group or a server-group to participate in an IP multicast addressing configuration for bridging over IP.
List	Displays the IP addresses of endstations participating in an IP multicast and unicast addressing configuration for bridging over IP. Also displays the size (in number of bytes) of bridging packets being routed through an IP tunnel and whether or not multicast addressing is enabled or disabled.
Set	Sets a new multicast IP address used for the IP multicast addressing configuration for bridging over IP.

Table 12 (Page 2 of 2). Tunnel Configuration Commands

Command	Function
Exit	Exits the tunnel configuration process and returns to the ASRT configuration process.

9.5.2 Configuration Commands for NetBIOS

Table 13 shows the list of the commands that you may use to configure NetBIOS parameters in the MSS Server.

Table 13. NetBIOS Configuration Commands

Command	Function
? (Help)	Lists all the NetBIOS configuration commands, or lists the options associated with specific commands.
Add	Adds cache entries to the router′s name cache.
Delete	Deletes cache entries add using the Add command.
Disable	Disables duplicate frame filtering and route caching.
Enable	Enables duplicate frame filtering and route caching.
List	Displays various cache entry and configuration information.
Set	Configures parameters for name caching, duplicate frame filtering and frame-type filtering.
Exit	Exits the NetBIOS configuration process and returns to the ASRT configuration process.

9.6 Bridge Filtering

The MSS Server provides a very extensive set of mechanisms to control the data being bridged:

- NetBIOS filtering

  NetBIOS filtering enables you to fine tune the NetBIOS traffic being bridged. NetBIOS filter items can be Exclude, meaning that matching data will be dropped, or Include, meaning data will be forwarded.

  Note: The definition of NetBIOS filters is much the same as MAC filtering described in 9.7, "MAC Filtering Feature (MCF)" on page 261.

- Address filtering

  Address filtering enables you to filter frames that contain specific source and/or destination MAC addresses. Per filter item, you indicate to which input and/or output port(s) it applies. Address filter items are always exclusive, meaning that they may cause data to be discarded.

- Protocol filtering

  Protocol filtering enables you to filter frames for a specific protocol. Per filter item, you indicate to which input and/or output port(s) it applies. Protocol filter items are always exclusive, meaning that they may cause data to be discarded.

- MAC filtering

The MAC filtering feature (MCF) provides you the most extensive method of controlling the data being bridged. MCF is extensively described in the following section.

9.7 MAC Filtering Feature (MCF)

This section provides a description of the MAC filtering feature (MCF) as well as an example of how to configure MCF using the Configuration Program.

9.7.1 Introduction

The MAC Filtering (MCF) feature allows you to specify filters to be applied during bridging processing. These filters are based on the source or the destination MAC address of each bridged packet.

The filters may be applied to one or more interfaces in either direction (input or output).

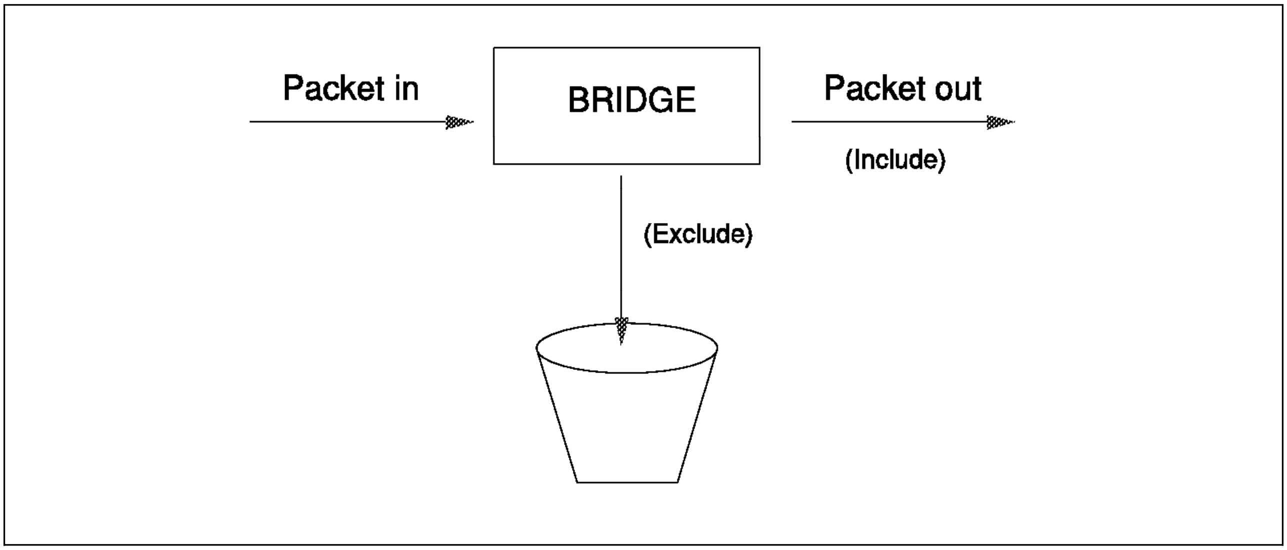


Figure 179. MAC Filtering Feature (MCF)

The action of a filter can be the following:

Include This means to bridge the packets normally.

Exclude This means to discard the packets.

Note: The current version of the configurator allows you to configure Tag as well. The operation of Tag is equivalent to Include.

Figure 179 shows the various actions supported by the MAC filtering feature.

The following is the description of the terminology used in the MAC filtering feature:

- A *filter-item* is a hexadecimal value that will be compared with the actual contents of a MAC frame being bridged. Filter-items contain the following:
  1. Filter-item value
     - Source MAC address
     - Destination MAC address
     - Sliding window

The MAC address filter-items are 6 bytes long and are always compared with either source or destination MAC address. A sliding window filter-item is of variable length. It is associated with an offset to indicate to which part of the MAC frame it will be compared.

2. Filter-item mask

   To reduce the number of filter-items that need to be defined a mask is associated with each value. Each zero bit in the mask field indicates a *don′t care* in the corresponding position of the filter-item value. The value and mask have the same length for each filter-item.

Examples are:

– (Source address=400012340000, Mask=FFFFFFFFFFFF) to indicate all MAC frames with source MAC address X′400012340000′.

– (Destination address=400012340000, Mask=4FFFFFFFFFFF) to indicate all MAC frames with either destination MAC address X′400012340000′ or X′C00012340000′.

  Note: Mask indicates that the top bit of the address is irrelevant.

– (Sliding window=400040,Offset=8,Mask=FF00FF), to indicate all MAC frames with X′40′ at offset 8 and 10.

- A *filter-list* is a group of one or more *filter-items*. Each filter-list has an associated action, which can be either Include or Exclude.
- A *filter* is a set of one or multiple filter-lists that are assigned to a specific bridge port. Filters are direction dependent, that is, a different filter is assigned for incoming (Input) and for outgoing (Output) traffic.

  Note: A filter-list can be part of multiple filters.

  Each input and output port for which filtering has been enabled has a default action associated: either Include or Exclude, which will be applied if none of the items in its filter apply.

Figure 180 on page 263 shows you the MAC filtering structure.

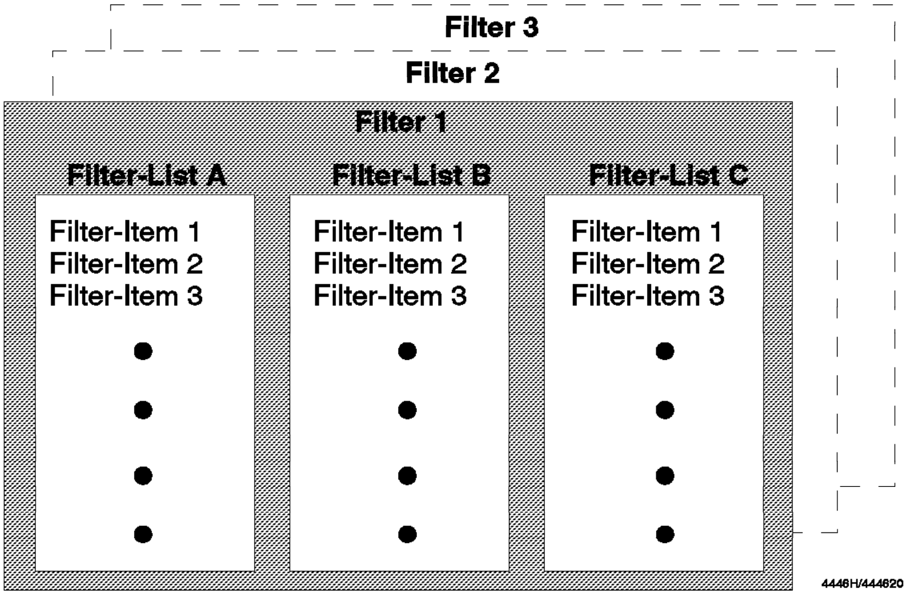


Figure 180. MAC Filtering Structure

The filter-items contained within a filter will be compared with the contents of each MAC frame entering (Input filter) or leaving (Output) the port to which a filter applies.

The order of the filter-items in a filter-list and the order of the filter-lists attached to a filter are very important. The following is a description of how the MAC filtering process works:

- The MAC filtering feature compares the MAC address in a packet with the filter-item number 1 in the filter-list number 1.
- If there is no match, the next filter-item (number 2) in the filter-list number 1 is checked. This process continues until the highest filter-item number in the filter-list number 1 has been checked.
- The above process is repeated with the next filter-list (number 2) and will continue until there is either a match with a filter-item or all the filter-items of all the filter-lists have been checked and no match has been found.
- As soon as a filter-item of a filter-list attached to the filter matches the packet′s MAC address, the MAC filtering feature performs the action of the filter-list to which the filter-item matching belongs.
- If no filter-item of the filter-lists attached to this filter matches the packet′s MAC address, the MAC filtering feature performs the default action of the filter.

Figure 181 on page 264 shows you the MAC filtering process for a *filter* with 3 filter-lists, each one with 2 filter-items.

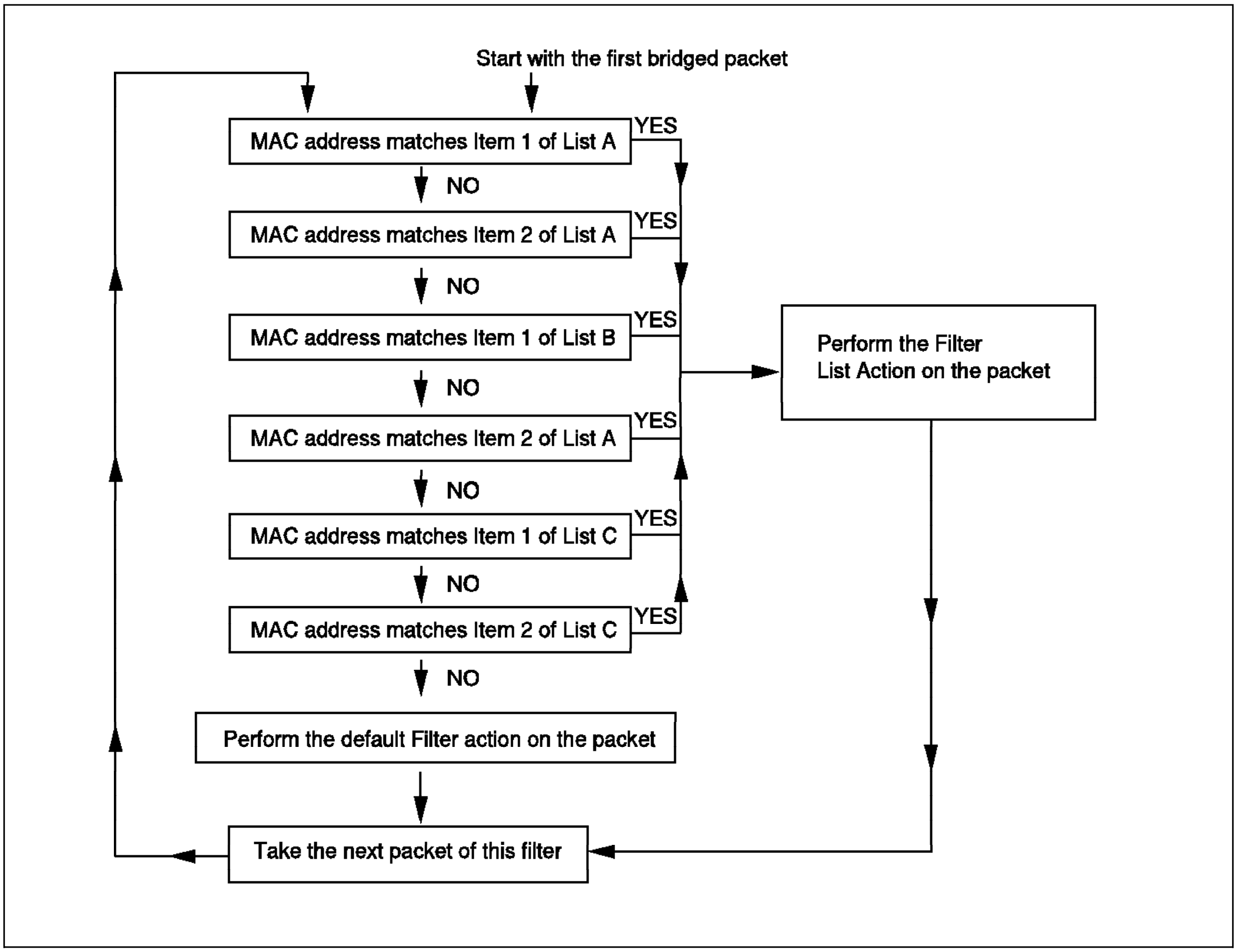

Start with the first bridged packet
MAC address matches Item 1 of List A
YES
NO
MAC address matches Item 2 of List A
YES
NO
MAC address matches Item 1 of List B
YES
NO
MAC address matches Item 2 of List A
YES
NO
MAC address matches Item 1 of List C
YES
NO
MAC address matches Item 2 of List C
YES
NO
Perform the default Filter action on the packet
Perform the Filter List Action on the packet
Take the next packet of this filter

Figure 181. MAC Filtering Process for a Specific Filter

9.7.2 MAC Filtering (MCF) Configuration Commands

This section summarizes the MAC filtering configuration commands available within the Config process of the MSS Server.

To access the MAC filtering feature you must enter the following command from the Config process:

```
Config>feature mcf
```

Table 14 on page 265 shows the available MAC filtering commands.

Table 15 on page 265 shows the available MAC filtering update subcommands.

The *MSS Server Command Line Interface Volume 1* covers these commands in detail.

Table 14. MAC Filtering Configuration Commands

Command	Function
? (Help)	Displays the MAC filtering commands, or lists subcommand options for specific commands (if available).
Attach	Attaches a filter-list to a filter.
Create	Creates a filter-list or a filter.
Default	Sets the default action for a filter with its specified filter number to Include or Exclude.
Delete	Deletes a filter-list with all its filter-item. Also deletes a filter previously created.
Detach	Detaches a filter-list from a filter.
Enable	Enables MAC filtering entirely or enables a specific filter.
List	Lists a summary of all the filters and filter-lists configured by the user. Also generates a list of attached filter-lists for this filter and all subsequent information for the filter.
Move	Reorders the *filter lists* attached to a specified filter.
Reinit	Reinitializes the entire MAC filtering system from an existing configuration without affecting the rest of the router.
Set-cache	Changes the cache size for a filter.
Update	Updates information for a specific filter-list. It brings you to a new prompt with a new menu of appropriate subcommands. Table 15 on page 265 shows you these new subcommands.
Exit	Exits the MAC filtering configuration environment and returns to the CONFIG process.

Table 15. MAC Filtering Update Subcommands

Command	Function
? (Help)	Displays all the Update subcommands.
Add	Adds filter-item to a filter-list. Adds a hexadecimal number with a hexadecimal mask to compare against the Source or Destination MAC address.
Delete	Deletes a filter-item from a filter-list.
List	Lists all the filter-items configured by the user for a filter-list. You can list either in canonical or non-canonical MAC address format.
Move	Reorders the filter-items attached to a filter-list.
Set-action	Sets the action for a filter-list to INCLUDE, EXCLUDE or TAG.
Exit	Exits the MAC filtering update subcommands environment and returns to MAC filtering commands.

9.8 Filtering - A Configuration Example

To illustrate how to configure bridge filters using the Configuration Program we show how to define an exclusive input filter for non-canonical source MAC address X′400011002200′ on the bridge port associated with logical interface 2.

The steps to configure the MAC filter feature are:

1. Create a filter-list

One or multiple filter-lists can be specified. Per filter-list define what the associated action is: Include or Exclude.

2. Add filter-items to the filter-list

   Per filter-list you can define multiple filter-items.

3. Enable filtering globally

   To use MAC filtering on any of the bridge ports requires that you enable filtering on box-level first.

4. Enable filtering per interface

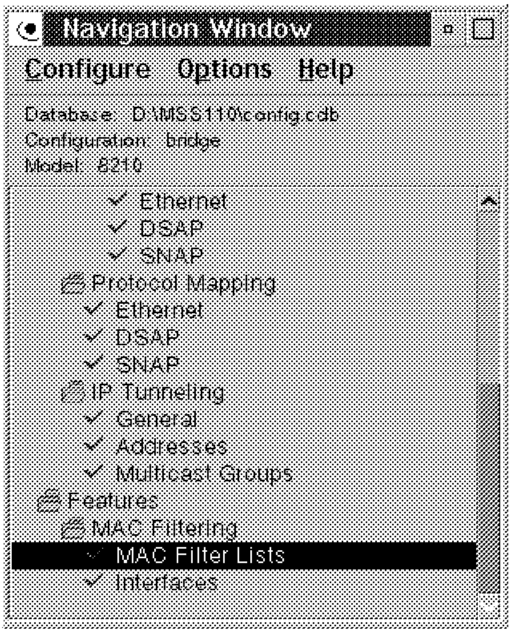
   Indicate on which bridge ports filtering has to be enabled. Note that this option is set seperately for incoming (Input) and/or outgoing (Output) data.

   For any of the input or output bridge ports on which filtering has been enabled, indicate what the default action will be (Include or Exclude), if none of the items within the filter-lists apply.

5. Assign the filter-list to the bridge port

   Assign one or multiple filter lists per bridge port. Note that this action is direction dependent and should be repeated for both incoming (Input) and outgoing (Output) data.

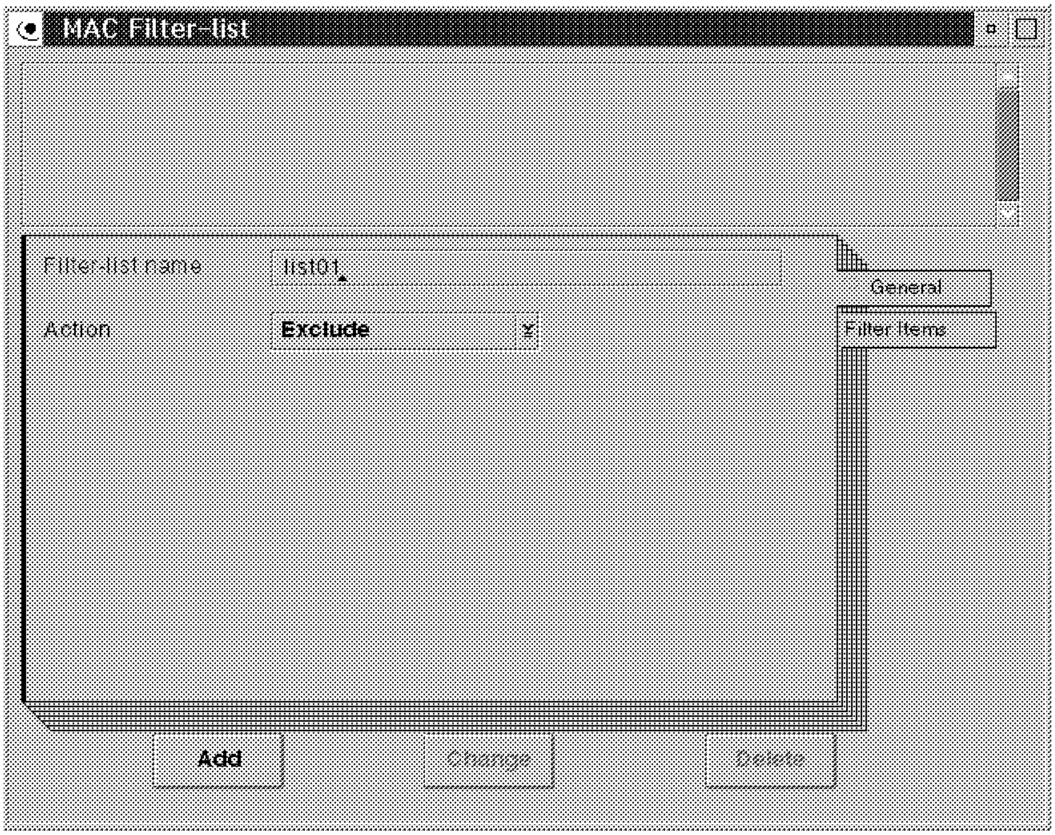
To start the definition process, select the MAC Filter Lists option in the Navigation Window, as shown in Figure 182.



Database: D:\MSS110\config.cdb
Configuration: bridge
Model: 8210

Figure 182. Select Filter Configuration

Figure 183 on page 267 appears.


MAC Filter-list

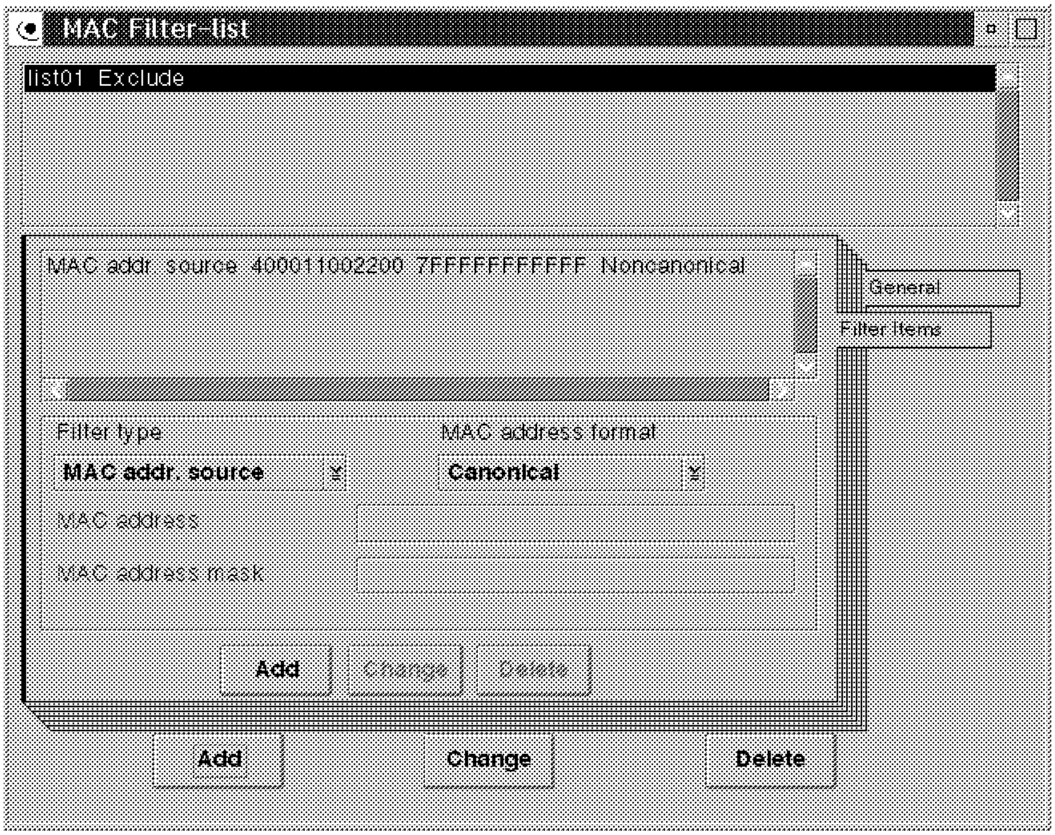
Filter-list name: list01

Action: Exclude

Figure 183. Create a Filter-List

Define a filter-list name (in our case *list01*) and indicate the action required (in our case exclude) for all filter-items that will be added.

The next step is to configure the filter-items within each of the filter-lists. To do so select Filter Items. Figure 184 on page 268 appears.



list01 Exclude

MAC addr source 400011002200 7FFFFFFFFFFF Noncanonical

Filter type: MAC addr. source

MAC address format: Canonical

Figure 184. Add Filter-Items

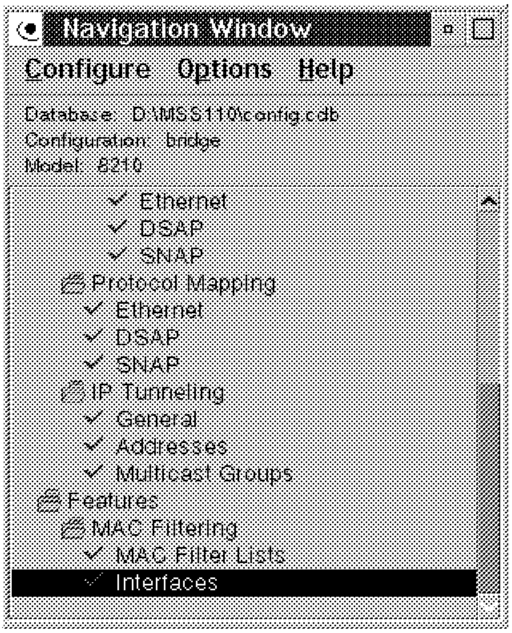
In our example, the only filter-item is the non-canonical source MAC address X′400011002200′. With this address we associate mask X′7FFFFFFFFFFF′. (As a result, source MAC address X′C00011002200′ will also be filtered.)

The filter type can be source MAC address, destination MAC address or sliding window. If the filter type is a MAC address, enter the type of address (canonical or non-canonical), the MAC address itself and an address mask. If the filter type is *sliding window*, enter the sliding window content, sliding window mask and the offset within the LAN frame.

Note: The use of sliding window enables filtering based on information anywhere in the MAC frame.

Selecting Add on the bottom line ends the configuration of MAC filter-list *list01*.

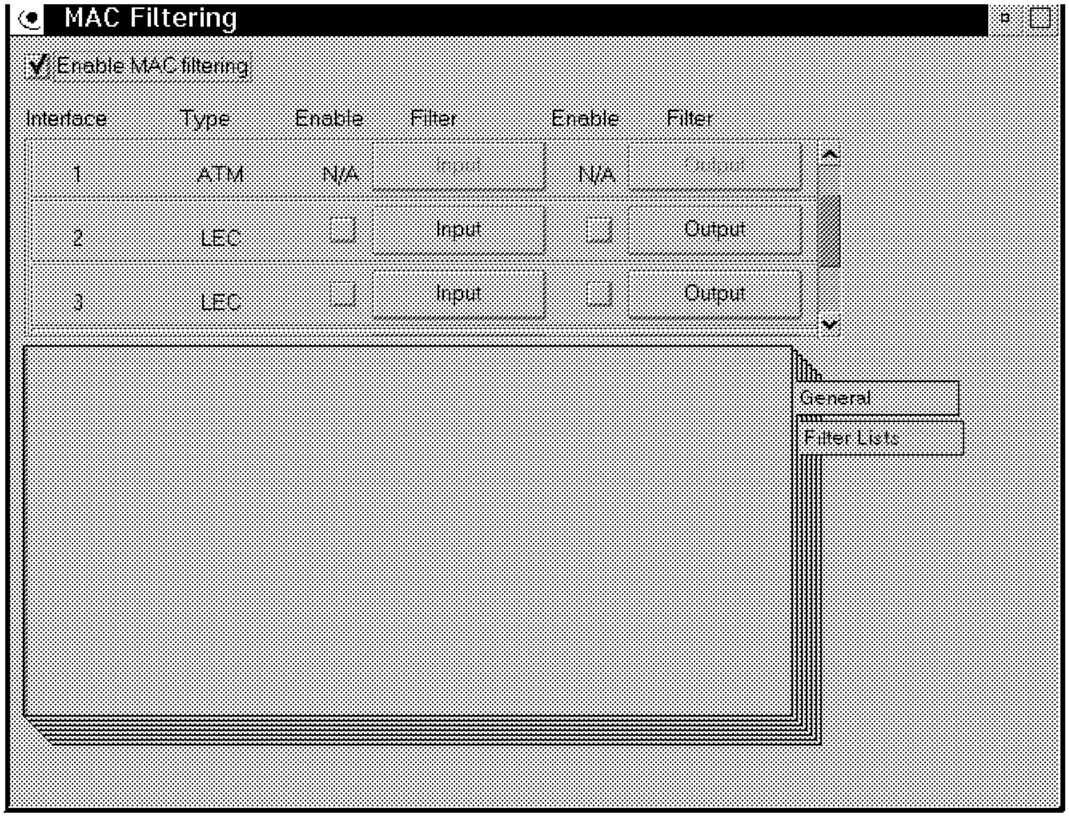
The next step is to apply the filter to the required bridging port. To do so select Interfaces in the MAC Filtering folder as shown in Figure 185 on page 269.



Database: D:\MSS110\config.cdb
Configuration: bridge
Model: 8210

Figure 185. Select Interfaces

Figure 186 results, showing all interfaces including the two token-ring logical interfaces.



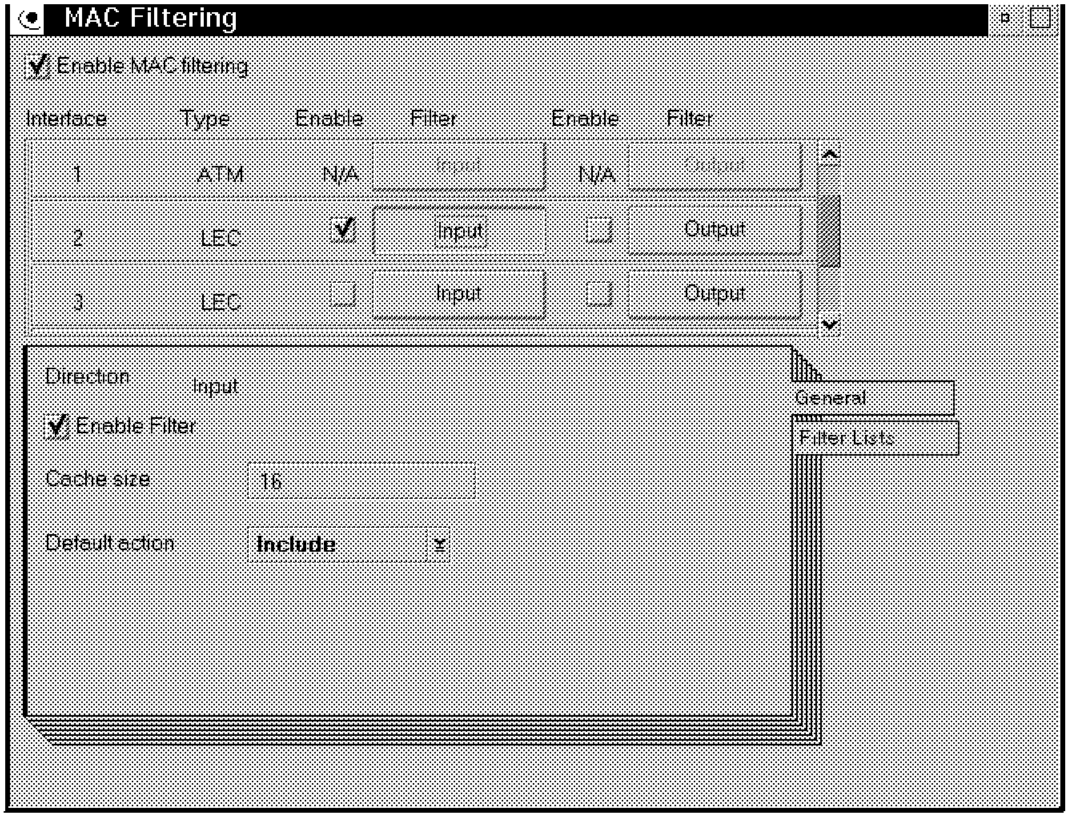
☑ Enable MAC filtering

Interface	Type	Enable	Filter	Enable	Filter
1	ATM	N/A	Input	N/A	Output
2	LEC	□	Input	□	Output
3	LEC	□	Input	□	Output

Figure 186. Configure Interfaces

Enable MAC filtering globally by clicking on the corresponding option at the top of the screen (see Figure 187 on page 270).

Thereafter, apply the bridge lists selectively to any of your logical interfaces on which MCF filtering is required. Indicate hereby if the bridge filters need to be applied to incoming data (select input) or to outgoing data (select output).



MAC Filtering

☑ Enable MAC filtering

Interface	Type	Enable	Filter	Enable	Filter
1	ATM	N/A	Input	N/A	Output
2	LEC	☑	Input	☐	Output
3	LEC	☐	Input	☐	Output

Direction Input

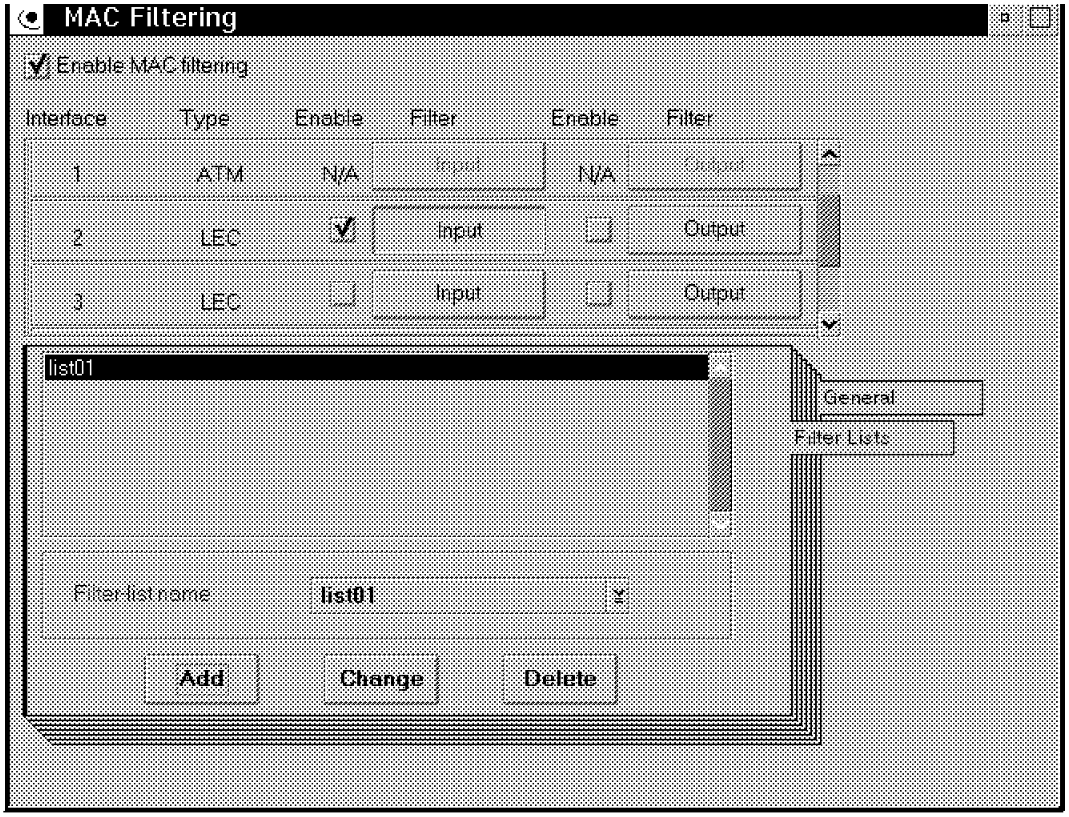
☑ Enable Filter

Cache size 16

Default action Include

Figure 187. Enable Filtering Globally

In our example, we enabled input filtering on logical interface 2. The default action, when none of filter item applies, is Include. By selecting Filter List we are able (see Figure 188 on page 271) to specify which filter-list (in our case *list01*) has to be used during the input filtering.



MAC Filtering

☑ Enable MAC filtering

Interface	Type	Enable	Filter	Enable	Filter
1	ATM	N/A	Input	N/A	Output
2	LEC	☑	Input	□	Output
3	LEC	□	Input	□	Output

list01

Filter list name: list01

Figure 188. Configure Interfaces

Selecting Add concludes our example. Note that multiple filter-lists applied to multiple logical interfaces can be defined.

Chapter 10. SNMP Management and Event Logging System (ELS)

Network management is a key element in today′s network environments. Its importance is increasing following the growth of network complexity.

In this chapter we provide an overview of the SNMP management functions and the monitoring functions offered by the event logging system (ELS) on the MSS Server.

For more detailed information on the SNMP management functions, see *Multiprotocol Switched Services (MSS) Server Configuration and Operations Guide*. More information about ELS can be found in the *Multiprotocol Switched Services (MSS) Server Command Line Interface Volume 1* and the *Multiprotocol Switched Services (MSS) Service Event Logging System Messages Guide*.

10.1 SNMP Management

The SNMP support for the MSS Server is based on the following:

1. SNMP Version 1 agent support

   The MSS Server provides an SNMP agent, which enables you to manage the MSS Server from any SNMP-capable network management station.

   The management functions supported depend on functions implemented on the network station. Most extensive support is available when using NWays Campus manager for AIX.

2. MIB support

   The MSS Server provides a comprehensive set of standard and enterprise-specific MIBs for monitoring and managing its resources. For an overview of the supported MIBs, see 10.1.2, "MIB Support" on page 275.

3. IP transport of SNMP messages between MSS Server and the network management stations

   Important

   It is important to realize that SNMP management requires IP connectivity between the management station and the MSS Server.

   IP connectivity can be in-band via the ATM network or out-of-band via a SLIP connection (see 2.1.2, "SLIP" on page 9). In-band connectivity requires an active MSS Server on which at least one IP address has been defined and an IP route to the management station.

10.1.1 SNMP Authentication

The SNMP agent in the MSS Server provides controlled access to all or part of its MIB data using the authentication methods defined in SNMP Version 1. On the MSS Server this involves the definition of an association between MIB views and SNMP communities.

A MIB view is a subset of the MIB information maintained. Zero, one or multiple MIB views can be defined. The MIB view (called *all*) encompassing the whole MIB database is predefined.

One or multiple SNMP communities must be defined, consisting of:

- Community name

  Each community is identified by its community name. The community name is used as a password when a network management station solicits MIB data (SNMP READ), sets MIB variables (SNMP WRITE), or during the receipt of unsolicited data (SNMP TRAP).

- Community members

  One or multiple IP hosts that are part of the community.

  Note: If no IP hosts are explicitly defined, all IP stations are considered to be a member of the community.

- Access privilege

  The access privilege granted to the members of the community. Options are:

  – TRAP only
  – READ TRAP
  – WRITE READ TRAP

- Type of TRAPS

  The type of TRAPS sent to the members of the community. Options are:

  – Cold start
  – Warm start
  – Link up
  – Link down
  – Authentication failure
  – EGP
  – Enterprise specific
  – All

  All except the enterprise specific traps are defined by RFCs. The enterprise specific traps are generated by event logging system (ELS) messages. Use the ELS `trap` command to enable sending of messages or groups of messages via an SNMP trap.

  Note: By not defining any type, no trap will be sent to any member within this community.

- Associated MIB view

  The MIB view associated with this community.

The definition of SNMP communities and associated MIB enables the MSS Server to validate solicited SNMP requests received from network management stations. Solicited SNMP requests will be accepted if all of the following conditions are fulfilled:

- The request contains a valid community name
- The network management station is a member of the community
- The SNMP request pertains to MIB variables that are part of the MIB view associated with the community
- The proper access privilege has been defined

Unsolicited SNMP data (TRAPs) will be sent to all network management stations for which TRAP access has been defined. The SNMP TRAP contains the community name that identifies the community in which the network management has been defined.

Note: The types of TRAPs that are sent to members within a specific community is configurable.

10.1.2 MIB Support

The MSS Server supports the following type of MIBs:

- Generic MIBs

  The generic MIB information is defined by the Proteon enterprise MIB. This MIB is used because the routing function of the MSS Server is based on Proteon software and covers the hardware-independent common software aspects of the MSS Server.

- Industry-standard MIBs

  The MIBs defined in RFCs.

- Enterprise-specific MIBs

  MIB information pertaining to a specific product.

- Other

  Enterprise-specific MIBs accepted as defacto standards, for example, the Novell IPX/RIP/SAP MIBs.

The list of supported MIBs include:

- RCF 1213 - Management Information Base (MIB II)
- RCF 1573 - Evolutions of the Interfaces Group of MIB II
- RFC 1253 - OSPF Version 2 MIB
- RFC 1493 - Bridge MIB
- RFC 1525 - Source Routing Bridge MIB
- RFC 1657 - BGP Version 4 MIB
- RFC 1695 - ATM MIB
- ATM_Forum/94-073R4 - LAN Emulation Server Management Specification 1.0, December 1995
- IBM LAN Emulation Server MIB Extensions
- Novell IPX MIB
- Novell RIP/SAP MIB
- ICMP Echo Request MIB
- Proteon Enterprise MIB
- Platform Specific MIBs

For more detailed information and possibly exceptions, see the *Multiprotocol Switched Services (MSS) Server Configuration and Operations Guide*.

10.1.3 SNMP Configuration Using Command Line

To enter SNMP configuration you have to enter the following command:

```
Config>protocol snmp
SNMP user configuration
SNMP Config>
```

You have the following command options to configure your SNMP protocol:

Table 16. SNMP Console Commands

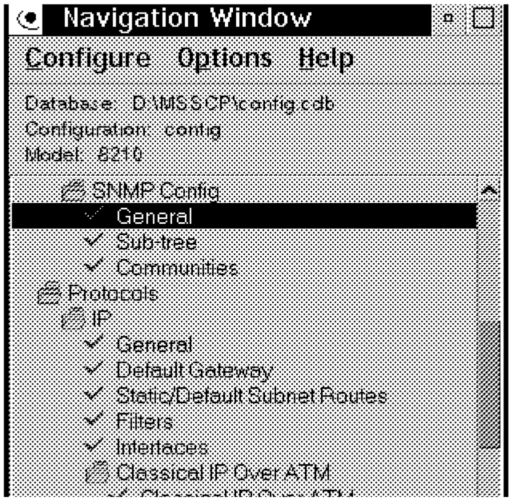
Command	Function
? (Help)	Lists the SNMP configuration commands or lists the options associated with specific commands.
Add	Adds a community to the list of SNMP communities, an IP address with mask to a community or a subtree to a MIB view.
Delete	Removes a community from the list of SNMP communities, an IP address with mask from a community or a subtree from MIB view.
Disable	Disables SNMP protocol and standard traps associated with named communities.
Enable	Enables SNMP protocol and standard traps associated with named communities.
Exit	Exits the SNMP configuration process and returns to the CONFIG environment.
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 an access node or view for a community and allows setting of trap UDP port. The access mode can be: • Read and trap generation • Read, write and trap generation • Trap generation only

10.1.4 SNMP Configuration Using the Configuration Program

We illustrate the procedure for configuring the SNMP functions using the Configuration Program by defining a simple SNMP community with:

- Community name: *mss_server*
- Community members: all hosts within subnet 9.24.104
- Access privilege: read-write trap
- Type of traps: all (traps to 9.24.104.110 only)
- Associated MIB view: all

To start our configuration we select General in the SNMP folder on the Navigation Window, as shown in Figure 189 on page 277.

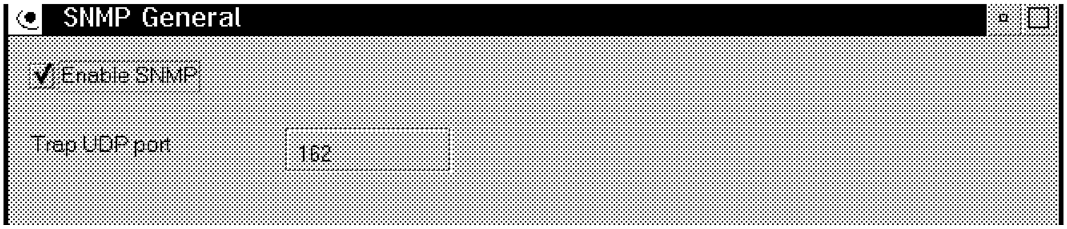


Database: D:\MSSCP\config.cdb
Configuration: config
Model: 8210

Figure 189. Configure SNMP

This results in Figure 190, allowing us to enable SNMP and define the trap UDP port used by SNMP.

Note: The UDP port should only be set if the default value of 162 is not appropriate.

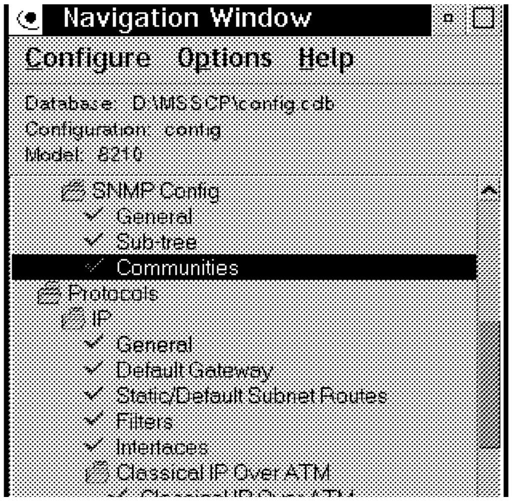


☑ Enable SNMP

Trap UDP port 162

Figure 190. Enabling SNMP

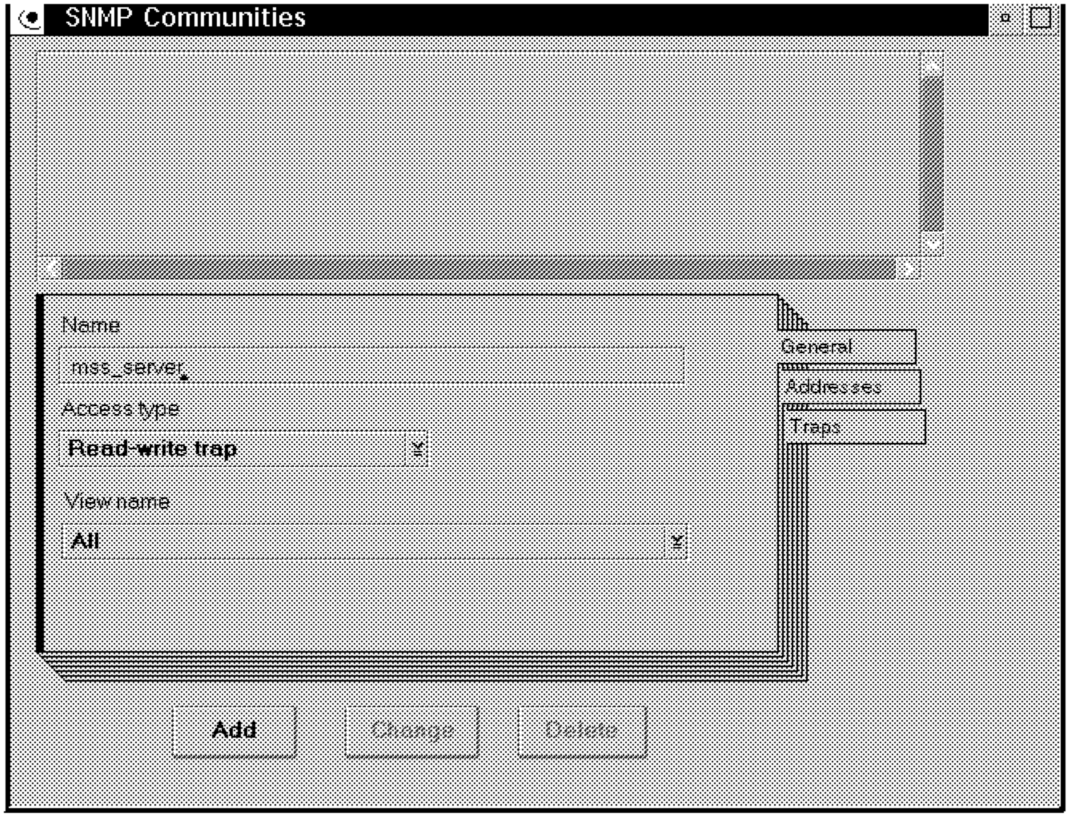
After enabling SNMP on box-level, its functions need to be configured. To do so select Communities in the Navigation Window as depicted in Figure 191 on page 278.



Database: D:\MSSCP\config.cdb
Configuration: config
Model: 8210

Figure 191. Configuring the Community

As a result Figure 192 appears.

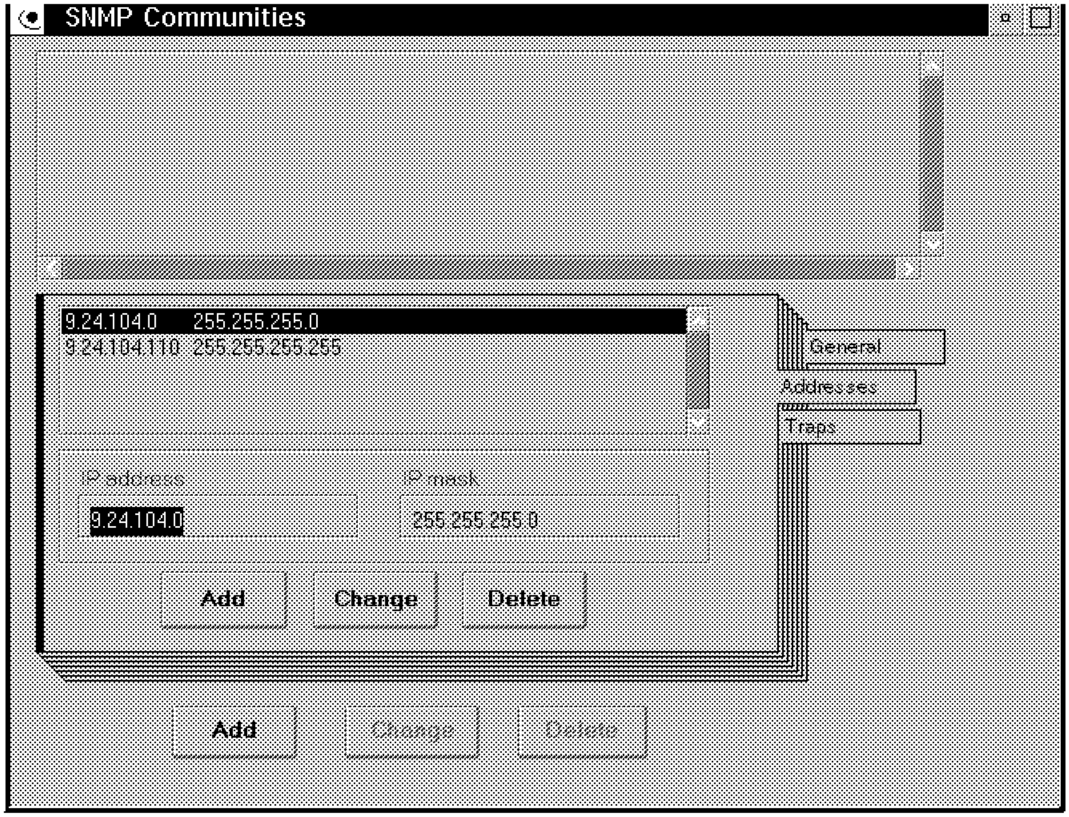


Name: mss_server
Access type: Read-write trap
View name: All

Figure 192. Configuring the Community

This menu enables you to define the community name, the access privilege, and the MIB view that is associated with this community. We did not select Add because we first wanted to define the members of *mss_server* community and define which type of TRAPs should be sent.

Selecting Addresses enables you to define the IP hosts that are part of community mss_server.



```
9.24.104.0     255.255.255.0
9.24.104.110   255.255.255.255
```

IP address: 9.24.104.0

IP mask: 255.255.255.0

Figure 193. Community Members

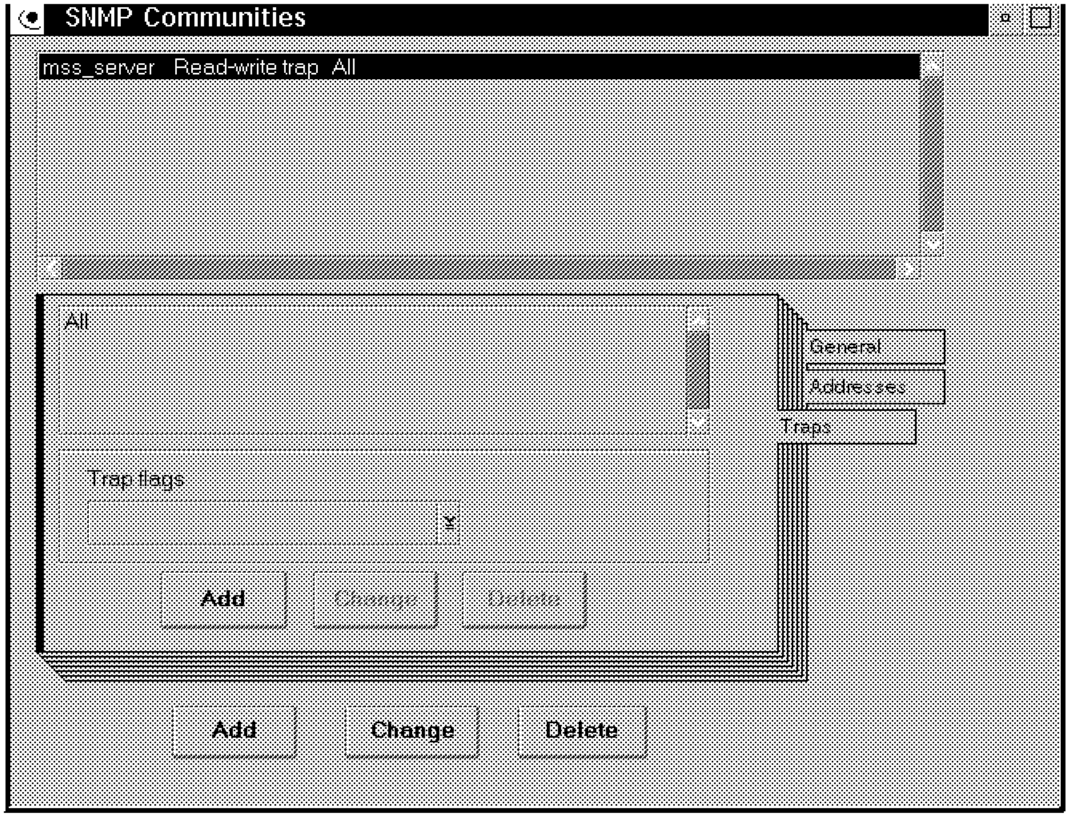
As displayed in Figure 193, the definition of the community members consists of specifying one or multiple sets of IP address(es) and IP mask(s).

Note: If no IP addresses are specified, all your IP hosts are considered to be a member of this community.

You can specify individual hosts by defining a valid host address and subnet mask 255.255.255.0 (see, for example, 9.24.104.110). A group of IP addresses can be defined by defining a subnet IP address and defining a subnet mask that is not all-ones (see, for example, 9.24.104.0).

SNMP traps will only be sent to IP hosts that have been specified with IP mask 255.255.255.255. As a result of the definitions in Figure 193:

- All hosts within subnet 9.24.104.0 have read/write access.
- Host 9.24.104.110 will receive SNMP traps as well.



mss_server Read-write trap All

All

Figure 194. Adding the Community

The type of traps that 9.24.104.110 receives can be configured by selecting Traps. Figure 194 appears. We select All to receive all traps generated by the MSS Server.

Adding the trap flag completed our SNMP configuration. By selecting Add on the bottom line, the SNMP configuration will be saved.

10.2 Event Logging System (ELS)

ELS is a monitoring system that manages messages logged as a result of the actions performed by the MSS Server. It is a powerful tool that can be used for monitoring purposes and for problem determination.

Using ELS commands, you can configure the system such that you only see the messages that you need. ELS uses the concepts of *subsystem, event number, message text, logging level, and group* to help you manage the messages you see.

Subsystem is a predefined name for a router component, such as an interface or protocol. For example, IP is the subsystem name for the IP protocol and ATM is the subsystem name for the ATM interface.

The ELS Config process is accessed by issuing the Config>**event** command. A complete list of the subsystem names can be obtained by issuing the ELS Config>**list subsystem** command as shown in Figure 195 on page 281.

The output shows the subsystem name, the number of events for the subsystem, and a description of the subsystem.

```
ELS config>list subsystem

Name  Events  Description

ALL           All subsystems
GW     98     Router base and network library
FLT     7     Filter Library
ARP    98     Address Resolution Protocol
IP    100     Internet Protocol
ICMP   21     Internet Control Message Protocol
TCP    57     TCP
UDP     4     User Datagram Protocol
BTP    13     BOOTP relay agent
RIP    19     IP Routing Information Protocol
OSPF   73     Open SPF-Based Routing Protocol
MSPF   15     OSPF Multicast extensions
TFTP   29     TFTP Protocol
SNMP   20     Simple Network Management Protocol
XN     21     XNS/IPX/DDS common processing
IPX    99     Internetwork Packet Exchange Protocol
SRT    89     Source Routing Transparent Bridge
STP    32     Spanning Tree Protocol
BR     30     Bridge/Routing
ETH    47     Ethernet Handler
TKR    45     Token Ring Handler
IPPN    4     IP Protocol Net
BGP    74     Border Gateway Protocol
MCF     9     MAC Filtering
NBS    50     NetBIOS Support Subsystem
ATM   165     Asynchronous Transfer Mode
LEC    43     ATM LAN Emulation Client
ILMI   23     ATM Interim Local Management Interface
SAAL   26     ATM Signalling ATM Adaptation Layer
SVC    25     ATM Signalling
LES   338     LAN Emulation Services
LECS  132     LAN Emulation Configuration Server
EVLOG    1     EventLog() error logging system
NOT    15     Forwarder messages not loaded
```

Figure 195. List Subsystem Command

Event number is a predefined number assigned to each message within a subsystem. You can obtain a complete list of events for a particular subsystem by issuing the ELS Config>**list subsystem** *subsys* command, where *subsys* is the name of the particular subsystem you are interested in; for example:

```
ELS Config>list subsystem lec
```

This command will list all possible events in the LEC subsystem, as shown in Figure 196 on page 282.

The output shows the event number, the logging level and the message text.

```
ELS config>list subsystem lec

Event      Level      Message

LEC.001    C-INFO     LEC function entry/exit tracing
LEC.002    C-INFO     nt %d %s %s
LEC.003    C-INFO     nt %d %s %s, D1=%d
LEC.004    C-INFO     nt %d %s %s, D1=%d, D2=%d
LEC.005    C-INFO     nt %d %s %s, D1=%d, D2=%d, D3=%d
LEC.006    C-INFO     nt %d %s %s, conn_handle=%d
LEC.007    C-INFO     nt %d %s %s, client_state=%s
LEC.008    C-INFO     reserved
LEC.009    U-INFO     nt %d LEC state chng from %s to %s
LEC.010    U-INFO     nt %d dest state chng from %s to %s
LEC.011    P-TRACE    Trace LEC data packet
LEC.012    P-TRACE    Trace LEC control packet
LEC.013    C-TRACE    nt %d Rcvd %s on conn handle %d with xid %x
LEC.014    C-TRACE    nt %d Sent %s on conn handle %d with xid %x
LEC.015    U-INFO     nt %d %s %s
LEC.016    U-INFO     nt %d %s %s, D1=%d
LEC.017    U-INFO     nt %d %s %s, D1=%d, D2=%d
LEC.018    U-INFO     nt %d %s %s, D1=%d, D2=%d, D3=%d
LEC.019    C-INFO     reserved
LEC.020    UE-ERROR   nt %d %s %s
LEC.021    UE-ERROR   nt %d %s %s, D1=%d
LEC.022    UE-ERROR   nt %d %s %s, D1=%d, D2=%d
LEC.023    UE-ERROR   nt %d %s %s, D1=%d, D2=%d, D3=%d
LEC.024    UI-ERROR   open frame SAP failed on nt %d, rc=%d
LEC.025    UI-ERROR   open call SAP failed on nt %d, rc=%d
LEC.026    UI-ERROR   open data path failed for outgoing call, on nt %d, rc=%d
LEC.027    UI-ERROR   open data path failed for incoming call, on nt %d, rc=%d
LEC.028    C-INFO     Function %S called, nt %d int %s/%d
LEC.029    UI-ERROR   Start failed, on nt %d int %s/%d, rc=%d
LEC.030    UI-ERROR   create LEC object failed, on nt %d int %s/%d, rc=%d
LEC.031    UI-ERROR   usr reg failed, on nt %d int %s/%d, rc=%d
LEC.032    UI-ERROR   nt %d int %s/%d, ATM nt %d int %s/%d nt nbld
LEC.033    UI-ERROR   LEC activate failed, on nt %d int %s/%d, rc=%d
LEC.034    UI-ERROR   LEC activate complete, on nt %d int %s/%d, rc=%d
LEC.035    UI-ERROR   Outbound frame freed, on nt %d int %s/%d
LEC.036    UI-ERROR   Outbound frame queued, on nt %d int %s/%d
LEC.037    UI-ERROR   Transmit failed, on nt %d int %s/%d, rc=%d
LEC.038    UI-ERROR   Outbound frame discarded, on nt %d int %s/%d, rsn=%d,state
=%d,hndl=%d
LEC.039    UI-ERROR   LEC inbnd fr dscrd, size %d, on nt %d int %s/%d
LEC.040    UI-ERROR   LEC inbnd fr dscrd, mcast addr, on nt %d int %s/%d
LEC.041    UI-ERROR   LEC inbnd fr dscrd, bad mac, on nt %d int %s/%d
LEC.042    UI-ERROR   SRAM nt fnd on dsbl, on nt %d int %s/%d
LEC.043    UI-ERROR   cancel alarm, on nt %d rc = %d, num %d
```

Figure 196. Output from the List Subsystem LEC Command

Message text is the actual text related to the event that has occurred and is used along with the subsystem and event number when the message is displayed by the MONITR process.

Logging level is a predefined category that each event will belong to, and which indicates the importance of the event. Note whenever you issue the `ELS Config>`**`list subsystem`** *subsys* command to list all the events within a subsystem, the logging level for each event is displayed.

The complete list of logging levels is shown in Table 17.

Table 17. Logging Levels

Level	**Description**
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 informational comment levels above
STANDARD	Includes all error levels and all informational comment levels above
PTRACE	Per packet trace
UTRACE	Unusual operation trace message
CTRACE	Common operation trace message
TRACE	Includes all trace levels above
ALL	Includes all logging levels

Group is a user-defined collection of events that is given a name. A group can consist of events from different subsystems and of different logging levels. Once you have created a group, you can use the group name to manipulate the events in the group as a whole.

The *Multiprotocol Switched Services (MSS) Server Event Logging System Messages Guide* contains a complete list of all events for all subsystems and includes the logging level for each event.

The configuration of ELS using the command line interface is started after entering the `event` command in CONFIG mode (see 2.3.6, "CONFIG Process" on page 27). The Config command prompt changes into `ELS Config>` and the ELS commands listed in Table 18 become available.

Table 18 (Page 1 of 2). ELS Configuration Commands

Command	**Function**
? (Help)	Lists the ELS configuration commands or lists the options associated with specific commands.
Add	Adds an event to an existing group or creates a new group.
Clear	Clears all ELS Configuration information.
Default	Resets the display or trap setting of an event, group, or subsystem.
Delete	Deletes an event number from an existing group or deletes an entire group.
Display	Enables message display on the console monitor (MONITR).
Exit	Exits the ELS configuration process.

Table 18 (Page 2 of 2). ELS Configuration Commands

Command	**Function**
List	Lists information on ELS settings and messages.
Nodisplay	Disables message display on the console.
Notrace	Disables trace event display on the console.
Notrap	Prevents messages from being trapped and sent out over SNMP.
Set	Sets the pin parameter, the time stamp feature and the trace status.
Trace	Enables trace event display on the console.
Trap	Allows messages to be trapped and sent out over SNMP.

ELS console commands can be entered after issuing the `event` command in GWCON mode (see 2.3.7, "GWCON Process" on page 30). The GWCON command prompt changes into `ELS>`, and the ELS commands listed in Table 19 become available.

Table 19. ELS Console Commands

Command	**Function**
? (Help)	Lists the ELS configuration commands or lists the options associated with specific commands.
Clear	Clear messages associated with specific events, groups or subsystems.
Display	Enables message display on the console.
Exit	Exits the ELS console process and returns to the GWCON prompt (+).
Files	Transfer trace files to another host on the network via TFTP.
List	Lists information on ELS settings and messages and trace status.
Nodisplay	Disables message display on the console.
Notrace	Disables trace event display on the console.
Notrap	Prevents messages from being trapped and sent out over SNMP.
Remove	Frees up memory by erasing stored information.
Retrieve	Reloads the saved ELS configuration.
Save	Stores the current configuration.
Set	Sets the pin parameter, the time stamp feature and the trace status.
Statistics	Displays available subsystems and pertinent statistics.
Trace	Enables trace event display on the console.
Trap	Allows messages to be trapped and sent out over SNMP.
View	Allows viewing of traced packets.

10.2.1 Controlling Message Display

To control the display of messages in MONITR, you need to use the Display and Nodisplay commands. These can be issued from within the following:

- The CONFIG process, where the changes will become part of the configuration when the MSS Server is reloaded
- The GWCON process, where the effect will be immediate, but will only last until the router is reloaded, unless you use the Save command to store the new configuration

For example, to display the standard level messages associated with the ATM subsystem, you would issue:

```
ELS>display subsystem ATM standard
```

and to prevent all ATM messages being displayed you would issue:

```
ELS>nodisplay subsystem ATM all
```

You would now need to switch to the MONITR process to see the effects of your command, after reloading if necessary.

10.2.2 Using ELS

To best use ELS, it is recommended that you do the following:

- Clearly define the problems and events you want to see in the MONITR process.
- Turn off all ELS messages by issuing the command Nodisplay subsystem all either in the CONFIG process (ELS Config> prompt) or in the GWCON process (ELS> prompt).
- Turn on only the messages related to the problem you try to resolve.
- Consult the *Messages Guide* to determine if the messages are normal or not.

If you choose the CONFIG process to set ELS parameters, this new configuration will be permanent as soon as you reload the MSS Server, otherwise if you choose to make your changes after the reload of the router using the GWCON process, these changes will take effect only at the moment and can be discarded at the next reload, unless you save it.

The next step is to monitor the information using the MONITR process, which is accessed when a *talk 2* command is issued at the OPCON prompt (*).

This brings you to a screen where all information related to the configured events will appear. This screen, with the messages, continually rotates through the router′s buffer. The following combination of keys can be used to stop and restart the displaying of messages:

- <Ctrl+S> to pause scrolling
- <Ctrl+Q> to resume scrolling
- <Ctrl+P> to return to the last process

You may want to capture the ELS output for future analysis. This can be done by starting a capture either via a Telnet connection or using a terminal emulator with this feature.

Performance Consideration

Be careful with the use of ELS. Especially tracing packets (see Table 17 on page 283 for logging levels) may have a negative impact on your network performance. Only use ELS for and during problem determination.

10.2.3 Event Monitoring - an Example

Figure 197 shows an example of the monitor screen.

```
*talk 2
SAAL.001: nt 0 Function sscop::cpaal_rx_data_indication entered
SAAL.001: nt 0 Function sscop::pdu_stat entered
SAAL.026: nt 0 recv status: 0000016D 0000019F 0B000195 00050026, len=12
SAAL.001: nt 0 Function sscop::txdata_handler entered
SAAL.002: nt 0 Function sscop::txdata_handler (no poll) extd
SAAL.002: nt 0 Function sscop::pdu_stat extd
SAAL.002: nt 0 Function sscop::cpaal_rx_data_indication extd
  GW.026: Mnt nt 1 int Eth/0
  GW.026: Mnt nt 5 int Eth/2
  GW.026: Mnt nt 6 int TKR/2
ILMI.008: nt0 recv Get
ILMI.003: nt0 ntrd func getf, state=ILMI_PREFIX_READY
ILMI.011: nt0 Snd GetRsp Vpi+Vci, state=ILMI_PREFIX_READY
ILMI.003: nt0 ntrd func BuildGetResp, state=ILMI_PREFIX_READY
ILMI.020: nt0 snt Get Response
 STP.022: Hello tmr exp TB-1
 STP.010: Sndg cfg BPDU TB-1 port 1, nt 1 int Eth/0
 LEC.015: nt 1 Debug LEC_PROCESS_OUTBOUND_FRAME
 LEC.016: nt 1 Debug LEC_SEND_MCAST_FRAME, D1=59
 LEC.014: nt 1 Sent TOPOLOGY_REQUEST on conn handle 55 with xid A00008E2
 STP.010: Sndg cfg BPDU TB-1 port 2, nt 2 int Eth/1
 LEC.015: nt 2 Debug LEC_PROCESS_OUTBOUND_FRAME
 LEC.016: nt 2 Debug LEC_SEND_MCAST_FRAME, D1=86
 LEC.014: nt 2 Sent TOPOLOGY_REQUEST on conn handle 75 with xid A00008DD
 STP.010: Sndg cfg BPDU TB-1 port 3, nt 3 int TKR/0
 LEC.015: nt 3 Debug LEC_PROCESS_OUTBOUND_FRAME
 LEC.016: nt 3 Debug LEC_SEND_MCAST_FRAME, D1=103
 STP.010: Sndg cfg BPDU TB-1 port 4, nt 4 int TKR/1
 LEC.015: nt 4 Debug LEC_PROCESS_OUTBOUND_FRAME
 LEC.016: nt 4 Debug LEC_SEND_MCAST_FRAME, D1=82
 STP.010: Sndg cfg BPDU TB-1 port 5, nt 5 int Eth/2
 LEC.015: nt 5 Debug LEC_PROCESS_OUTBOUND_FRAME
 LEC.016: nt 5 Debug LEC_SEND_MCAST_FRAME, D1=202
 LEC.014: nt 5 Sent TOPOLOGY_REQUEST on conn handle 200 with xid A000089A
 STP.028: Attmpt root TB-1, strt hello tmr
 LEC.002: nt 1 Func entry process_inbound_ctl_frame
 LEC.013: nt 1 Rcvd TOPOLOGY_REQUEST on conn handle 58 with xid A00008E2
 LEC.004: nt 1 Func entry lec::CLSM_topology_request, D1=7, D2=0
 LEC.003: nt 1 Func exit  lec::CLSM_topology_request, D1=0
 LEC.002: nt 1 Func exit  process_inbound_ctl_frame
 LEC.015: nt 1 Debug LEC_PROCESS_INBOUND_MCAST_FRAME
 LEC.002: nt 2 Func entry process_inbound_ctl_frame
 LEC.013: nt 2 Rcvd TOPOLOGY_REQUEST on conn handle 83 with xid A00008DD
 LEC.004: nt 2 Func entry lec::CLSM_topology_request, D1=7, D2=0
 LEC.003: nt 2 Func exit  lec::CLSM_topology_request, D1=0
 LEC.002: nt 2 Func exit  process_inbound_ctl_frame
 LEC.015: nt 2 Debug LEC_PROCESS_INBOUND_MCAST_FRAME
 LEC.015: nt 3 Debug LEC_PROCESS_INBOUND_MCAST_FRAME
 LEC.015: nt 4 Debug LEC_PROCESS_INBOUND_MCAST_FRAME
 LEC.015: nt 5 Debug LEC_PROCESS_INBOUND_MCAST_FRAME
```

Figure 197. MONITR - an Example

In this case we can see messages from the ILMI, LEC, STP, GW and SAAL subsystems.

Chapter 11. Implementation Scenarios

This chapter contains a large number of implementation scenarios that can be used to build emulation LANs and/or Classical IP subnets using the IBM 8210 Nways MSS Server. We have tried to avoid confusion by using simple examples that are to a large extent self-contained, helping the reader to avoid having to page back and forth. In 11.5, "Putting Things Together" on page 452 we discuss a more complex scenario and show how it can be built using the simple scenarios introduced before.

11.1 Implementing the Scenarios

All scenarios start with a logical overview of the environment to be built. After the logical view a picture is shown of the relevant 8210 parameters. For simplicity we keep all remaining parameters at their default value. After displaying the logical layout and the relevant parameters, we show how the 8210 has been configured using the command line configurator. Note that for your convenience all keyboard input is printed in bold.

11.2 Design Considerations

It should be realized that when building any networking construct, ranging from simple to very complex, requires you to specify details on the following basic building blocks:

- ATM port definition
- LE client (token-ring, Ethernet)
- LES/BUS (possibly including BCM)
- LECS
- Classical IP client
- Classical IP (ARP) server
- Bridging (source-route, transparent, source-route to transparent)
- IP routing (possibly using dynamic routing protocols)
- IPX routing
- IP host services
- SNMP

Before starting to configure the IBM 8210 Nways MSS Server you should analyze your networking requirements and indicate:

1 How many ATM ports have to be configured and what UNI version is being used on each of them?

2 Is the IBM 8210 Nways MSS Server employed in LAN emulation, and if yes:

a. How many LE clients are used, of which type (token-ring or Ethernet), and using which ATM physical interface?

Note: Reasons to define LE clients on the IBM 8210 Nways MSS Server are:

- You need IP connectivity using ATM LAN emulation to the IBM 8210 Nways MSS Server to access its configuration and management functions

- You are using the IBM 8210 Nways MSS Server for IP routing functions between ELANs or between Classical IP subnets and ELAN(s)
- You are using the IBM 8210 Nways MSS Server for bridging functions
- You are using the IBM 8210 Nways MSS Server for IPX routing functions

b. Are the IBM 8210 Nways MSS Server LES/BUS functions being used, for which ELANs, and are the BCM functions being used?

If LES/BUS is being used:

- Is LECS/LES security feature being used?

If BCM is being used:

- For which protocols (NetBIOS, IP, IPX), and is source-route management (SRM) enabled?

c. Are the IBM 8210 Nways MSS Server LECS functions being used, for which ELANs, what are the policies used to assign clients to specific ELANS, and are LES/BUS′s internal or external?

3 Is the IBM 8210 Nways MSS Server employed in Classical IP, and if yes:

- How many Classical IP clients are required, and using which ATM physical interface?

  Note: Reasons to define Classical IP clients on the IBM 8210 Nways MSS Server are:

  – You need IP connectivity using Classical IP to the IBM 8210 Nways MSS Server to access its configuration and management functions

  – You are using the IBM 8210 Nways MSS Server for IP routing functions between Classical IP subnets and/or Classical IP subnets and ELAN(s)

- How many Classical IP servers are required and using which ATM physical interface?

4 Do you need the bridging functions on the IBM 8210 Nways MSS Server? If yes, what type: source route, transparent, or source-route to transparent?

5 Do you need the IP routing functions on the IBM 8210 Nways MSS Server, and which, if any, of the dynamic routing protocols are being used (RIP, OSPF, and/or BGP)?

6 Do you need the IPX routing functions on the IBM 8210 Nways MSS Server?

7 Do you need the IP host services on the IBM 8210 Nways MSS Server?

Note: Reasons to define the IP host services on the IBM 8210 Nways MSS Server are:

- The IBM 8210 Nways MSS Server is not used for IP functions (that is, no IP address has been assigned to it).
- IP access is required to the IBM 8210 Nways MSS Server to enable ATM (*in-band*) access to its configuration and management functions.

8 Do you need SNMP functions?

SNMP functions are required to enable the MSS Server to be managed from a network management stations, and/or to enable workstations running the Configuration Program to retrieve and store configuration files.

Once the basic building blocks for your environment have been identified, you can start configuring using the following examples.

11.3 Configuration Examples - Overview

The following sections list and briefly discuss a great number of configuration examples. The detailed configuration steps required for each scenario can be found in 11.4, "Configuration Examples - Details" on page 294.

11.3.1 ATM Attachment

Configuration scenarios described are:

1. Connect IBM 8210 Nways MSS Server to IBM 8260 using UNI auto-detection

   This scenario describes the basic configuration steps required to provide an ATM connection to an adjacent ATM switch using auto-detection of the UNI version on the adjacent ATM switch.

   For details see 11.4.1, "IBM 8210 ATM Attachment with UNI Auto-Detection" on page 295.

2. Connect IBM 8210 Nways MSS Server to IBM 8260 using predefined UNI

   This scenario describes the basic configuration steps required to provide an ATM connection to an adjacent ATM switch using a preconfigured UNI version.

   For details see 11.4.1, "IBM 8210 ATM Attachment with UNI Auto-Detection" on page 295.

11.3.2 LAN Emulation

Configuration scenarios described are:

1. IBM 8210 token-ring LE Client, external LES/BUS, no LECS

   This scenario describes the configuration steps required to define a token-ring LE client using an external LES/BUS without the LECS function active (that is, LES ATM address predefined on the LE client).

   For details see 11.4.3, "IBM 8210 Token-Ring LE Client" on page 303.

2. IBM 8210 Ethernet LE Client, external LES/BUS, no LECS

   This scenario describes the configuration steps required to define an Ethernet LE client using an external LES/BUS without the LECS function active (that is, LES ATM address predefined on the LE client).

   For details see 11.4.4, "IBM 8210 Ethernet LE Client" on page 307.

3. IBM 8210 token-ring LE Client, external LES/BUS, with LECS

   This scenario describes the configuration steps required to define a token-ring LE client using an external LES/BUS and an active (internal) LECS from which the LES address is obtained.

To enable the LE client to learn the LECS address dynamically, some configuration on the adjacent ATM switch is required. For details see 11.4.5, “IBM 8210 Token-Ring LE Client with LECS” on page 310.

4. IBM 8210 token-ring LE Client, internal LES/BUS, internal LECS, no LECS/LES security

   This scenario describes the configuration steps required to define a token-ring LE client using an internal LES/BUS and LECS.

   To enable the LE client to learn the LECS address, some configuration on the adjacent ATM switch is required. Note that in this scenario the LECS/LES security feature is not used and LE clients can join the ELAN without registering at the LECS first.

   For details see 11.4.6, “IBM 8210 Token-Ring LE Client, LES/BUS and LECS” on page 316.

5. IBM 8210 Ethernet LE Client, internal LES/BUS, and internal LECS, LECS/LES security enabled

   This scenario describes the configuration steps required to define an Ethernet LE client using an internal LES/BUS and internal LECS. To make sure only authorized LE clients connect to the ELAN (that is, verified by the LECS), the LECS/LES security interface is enable, and the ELAN has been defined as a secure ELAN.

   To enable the LECS/LES security feature to learn the LECS address, some configuration on the adjacent ATM switch is required. For details see 11.4.7, “IBM 8210 Ethernet LE Client, LES/BUS, and LECS” on page 324.

11.3.3 Classical IP

Configuration scenarios described are:

1. LIS client

   This scenario describes the configuration steps required to define a LIS client.

   For details see 11.4.8, “IBM 8210 LIS Client” on page 332.

2. LIS client/server

   This scenario describes the configuration steps required to define a LIS client/server.

   For details see 11.4.9, “IBM 8210 ARP Server” on page 336.

3. LIS client to LIS client using PVC

   This scenario describes the configuration steps required to define a LIS client connecting to another LIS client using a PVC.

   For details see 11.4.10, “IBM 8210 LIS Client to LIS Client Using PVC” on page 340.

4. LIS client to LIS client using predefined SVC

   This scenario describes the configuration steps required to define a LIS client connecting to another LIS client using an SVC, without defining an ARP server.

   For details see 11.4.11, “IBM 8210 LIS Client to LIS Client Using SVC, no ARP Server” on page 346.

11.3.4 IP Routing

Configuration scenarios described are:

1. LIS to LIS

   This scenario describes the configuration steps required to define two LIS clients. Note that the IBM 8210 Nways MSS Server is enabled by default for IP routing when multiple LIS clients have been defined.

   For details see 11.4.12, "IBM 8210 LIS to LIS Routing" on page 351.

2. LIS to token-ring ELAN

   This scenario describes the configuration steps required to define a LIS client and a token-ring LE client and to enable IP routing between them.

   For details see 11.4.13, "IBM 8210 LIS to Token-Ring ELAN IP Routing" on page 356.

3. Token-ring ELAN to Ethernet ELAN

   This scenario describes the configuration steps required to define two LE clients, Ethernet and token-ring respectively, and to enable IP routing between them.

   For details see 11.4.14, "IBM 8210 Ethernet to Token-Ring ELAN IP Routing" on page 362.

11.3.5 IPX Routing

Configuration scenarios described are:

1. Token-ring ELAN to Ethernet ELAN

   This scenario describes the configuration steps required to define two LE clients, Ethernet and token-ring respectively, and to enable IPX routing between them.

   For details see 11.4.15, "IBM 8210 Token-Ring to Ethernet ELAN IPX Routing" on page 373.

11.3.6 Bridging

Configuration scenarios described are:

1. Source-Route Bridging

   This scenario describes the configuration steps required to define two token-ring LE clients and to enable source-route bridging (SRB) between them.

   For details see 11.4.16, "IBM 8210 Source-Route Bridging (SRB)" on page 384.

2. Transparent Bridging

   This scenario describes the configuration steps required to define two Ethernet LE clients and to enable transparent bridging (TB) between them.

   For details see 11.4.17, "IBM 8210 Transparent Bridging" on page 397.

3. Source-Route Translational Bridging

   This scenario describes the configuration steps required to define two LE clients, token-ring and Ethernet respectively, and to enable source-route to transparent bridging (SR-TB) between them.

For details see 11.4.18, “IBM 8210 Source Route Translational Bridging (SR-TB)” on page 410.

11.3.7 SNMP Scenario

This scenario describes the configuration steps required to provide read and write access to the SNMP variables maintained on the MSS Server and to send SNMP traps to a network management station.

For details see 11.4.19, “SNMP Functions” on page 423.

11.3.8 Redundancy Scenarios

Configuration scenarios described are:

1. LECS and LES/BUS Redundancy

   This scenario describes the configuration steps required to configure an emulated LAN for which a primary and a backup LECS and a primary and backup LES/BUS have been defined.

   For details see 11.4.20, “LECS and LES/BUS Redundancy” on page 425.

2. ARP Server Redundancy

   This scenario describes the configuration steps required to configure a Classical IP subnet in which a primary and a backup ARP server have been defined.

   For details see 11.4.21, “ARP Server Redundancy” on page 442.

3. IP Gateway Redundancy

   This scenario describes the configuration steps required where two 8210 LIS clients provide redundant default IP gateway support for other LIS clients and LE clients.

   For details see 11.4.22, “IP Gateway Redundancy” on page 443.

4. Spanning Tree Root Bridge Redundancy

   This scenario describes the configuration steps required where two 8210s provide redundant spanning tree root bridge support.

   This configuration has been discussed in 5.10, “Redundant Spanning Tree Root Bridge” on page 118. For configuration details, see 11.4.23, “Spanning Tree Root Bridge Redundancy” on page 445.

11.4 Configuration Examples - Details

The following sections give details on the configuration effort required to implement the configuration examples listed in 11.3, “Configuration Examples - Overview” on page 291.

11.4.1 IBM 8210 ATM Attachment with UNI Auto-Detection

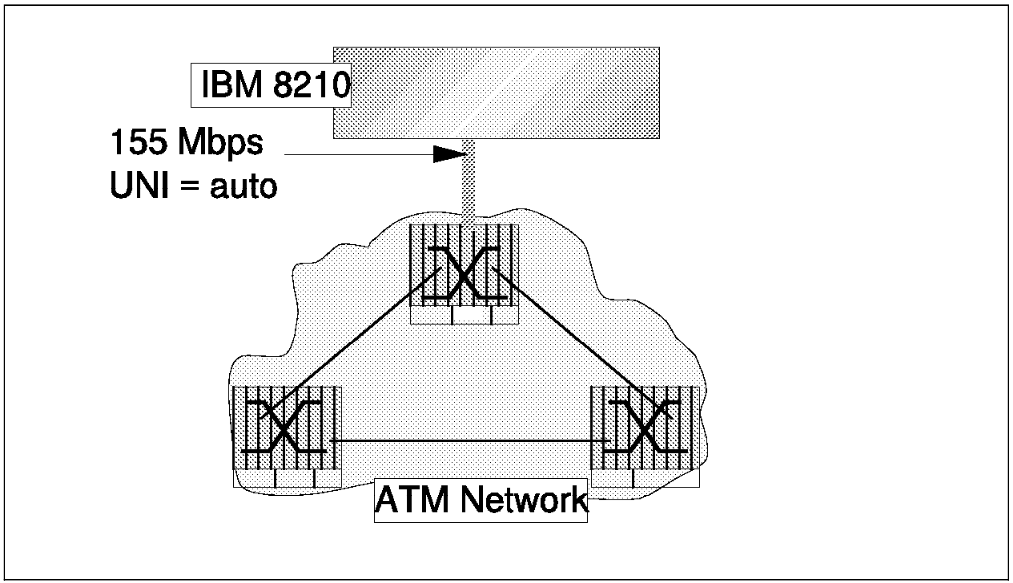


Figure 198. Auto-Detected UNI

Figure 198 shows the connection of the 8210 to the ATM network. The 8210 is capable of learning the UNI version supported at the switch′s port. For administrative and problem-solving reasons it is recommended that you use locally administered ESIs.

Corresponding UNI definitions are required on the adjacent ATM switch. When defining an LECS on the 8210 make sure that its ATM address is entered in the ILMI MIB of all ATM switches to which LE clients connect.

Note: The 20-byte LECS ATM address is comprised of the 13-byte network prefix, the 6-byte (locally administered) ESI, and selector byte 0. The network prefix can be learned from the ATM switch.

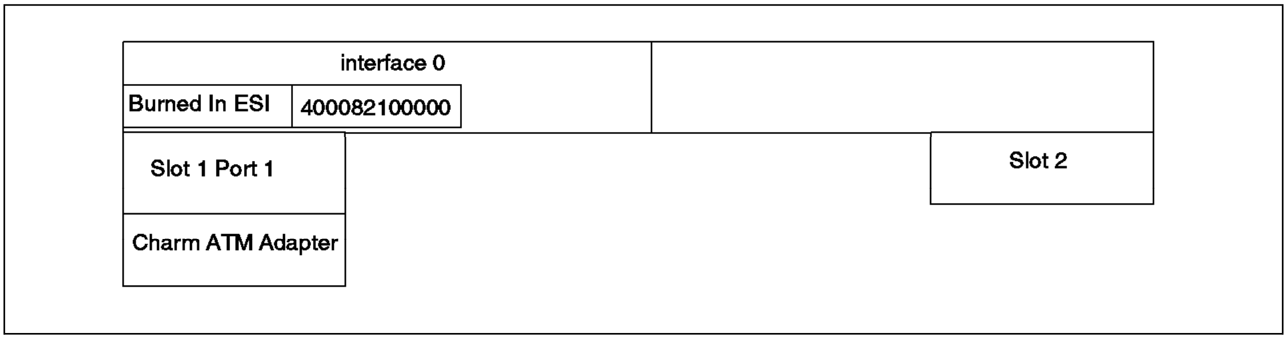


Figure 199. ATM Interface - Parameter Overview

To realize the scenario depicted in Figure 198 we used the parameters depicted in Figure 199. For the parameters not shown we used default value. The configuration steps required are:

1 **Configure the ATM interface**

Set UNI version

```
*talk 6

Config>list devices
Ifc 0 CHARM ATM PCI Adapter           Slot: 1  Port: 1
Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version
UNI version [3.0, 3.1, AUTO] []? AUTO
ATM Interface Config>
```

Figure 200 (Part 1 of 3). ATM Interface Configuration

List interface 0

```
ATM Interface Config>list configuration

                         ATM Configuration

  Interface (net) number =    0
  Maximum VCC data rate Mbps   =     155
  Maximum frame size     = 9234
  Maximum number of callers =  209
  Maximum number of calls = 1024
  Maximum number of parties to a multipoint call =  512
  Maximum number of Selectors that can be configured  =  200
  UNI Version = AUTO
  Packet trace = OFF
ATM Interface Config>
```

Figure 200 (Part 2 of 3). ATM Interface Configuration

Add and list locally administered ESIs

```
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>list esi

          ESI         Enabled
  -----------------   -------
  40.00.82.10.00.00       YES
ATM Interface Config>
```

Figure 200 (Part 3 of 3). ATM Interface Configuration

2 Configure the adjacent ATM switch

Obtain network prefix (first 13 bytes of ATM address)

```
8260ATM1> show device
8260 ATM Control Point and Switch Module
Name : 8260A
Location :
ITSO Raleigh, B678

For assistance contact :
Jaap de Goede

Manufacture id: VIME
Part Number: 58G9605 EC Level: C38846
Boot EEPROM version: v.1.2.0
Flash EEPROM version: v.2.0.4
Flash EEPROM backup version: v.2.0.4
Last Restart : 12:37:21 Fri 20 Sep 96 (Restart Count: 66)

A-CPSW
-------------------------------------------------------------------------------
 ATM address: 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.82.60.A1.00
```

Figure 201 (Part 1 of 2). 8260 Switch Configuration

Set the LECS well-known address (ILMI MIB entry)

```
8260ATM1> set lan_emul configuration_server
Enter WKA activity: inactive_wka
Enter ATM address : 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00
Entry set.
8260ATM1> show lan_emul configuration_server
Index           ATM address
-------------------------------------------------------------------------
 1           39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00
8260ATM1>
```

Figure 201 (Part 2 of 2). 8260 Switch Configuration

Important

The definition of the LECS address is irrelevant for this scenario but has been added for the scenarios which require LECS functions.

3 Restart 8210 and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 202 (Part 1 of 2). Restart and Verify

Display Interface Status

```
*talk 5

CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list addresses

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
  No Addresses Registered
ATM Interface+list vccs

 Conn    Conn                         Frames       Frames     Bytes        Bytes

 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R′cvd
 ------  ----  ---  ---- -----------  -----------  ---------  -----------  -----
      2  ILMI    0    16           0          51          50        2623    3231
ATM Interface+
```

Figure 202 (Part 2 of 2). Restart and Verify

11.4.2 IBM 8210 ATM Attachment with Preconfigured UNI

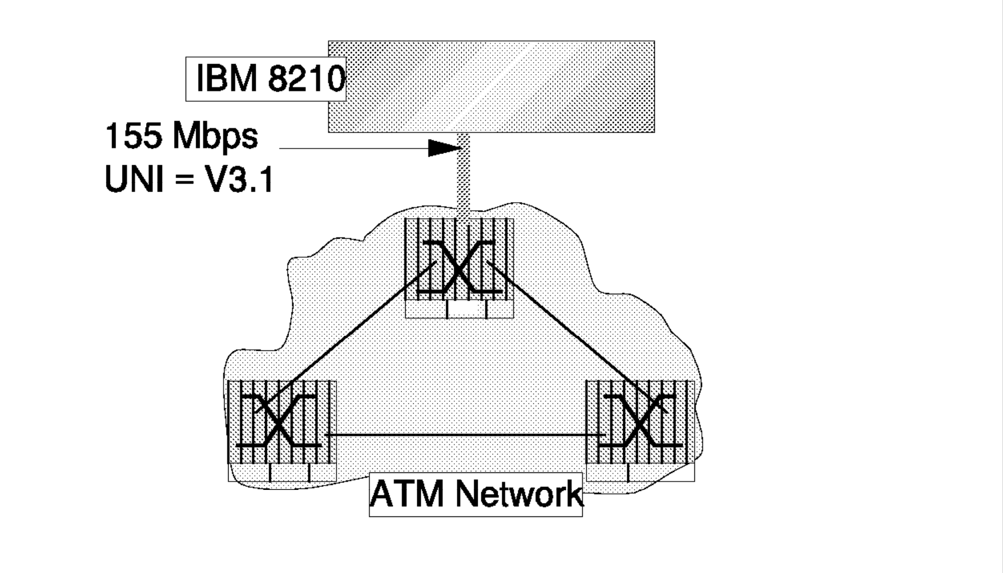


Figure 203. Preconfigured UNI

Figure 203 shows the connection of the 8210 to the ATM network. The 8210 is preconfigured to use UNI Version 3.0. For administrative and problem-solving reasons, it is recommended that you use locally administered ESIs.

Corresponding UNI definitions are required on the adjacent ATM switch. When defining an LECS on the 8210 make sure that its ATM address is entered in the ILMI MIB of all ATM switches to which LE clients connect.

Note: The 20-byte LECS ATM address is comprised of the 13-byte network prefix, the 6-byte (locally administered) ESI, and selector byte 0. The network prefix can be learned from the ATM switch.

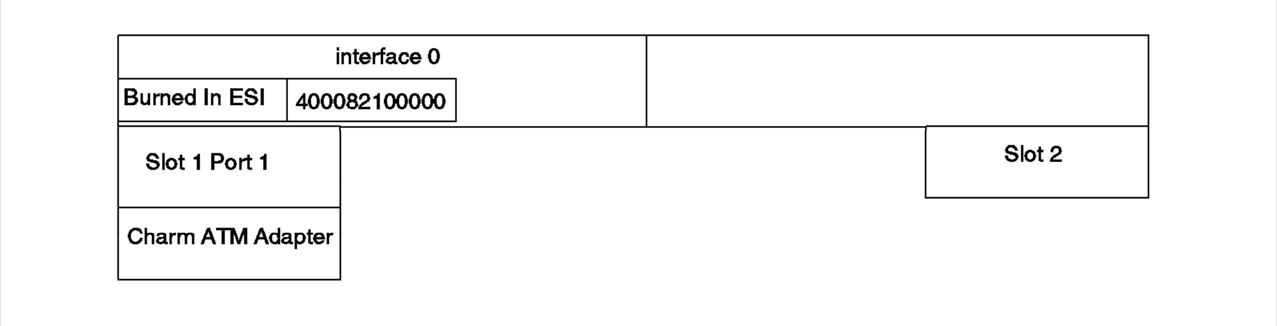


Figure 204. ATM Interface - Parameter Overview

To realize the scenario depicted in Figure 203 we used the parameters depicted in Figure 204. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

Set UNI version

```
*talk 6

Config>list devices
Ifc 0 CHARM ATM PCI Adapter           Slot: 1  Port: 1
Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version
UNI version [3.0, 3.1, AUTO] []? 3.1
ATM Interface Config>
```

Figure 205 (Part 1 of 3). ATM Interface Configuration

List interface 0

```
ATM Interface Config>list configuration

                         ATM Configuration

  Interface (net) number =    0
  Maximum VCC data rate Mbps   =     155
  Maximum frame size     = 9234
  Maximum number of callers =  209
  Maximum number of calls = 1024
  Maximum number of parties to a multipoint call =  512
  Maximum number of Selectors that can be configured  =  200
  UNI Version = UNI 3.1
  Packet trace = OFF
ATM Interface Config>
```

Figure 205 (Part 2 of 3). ATM Interface Configuration

Add and list locally administered ESIs

```
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>list esi

          ESI         Enabled
  -----------------   -------
  40.00.82.10.00.00       YES
ATM Interface Config>
```

Figure 205 (Part 3 of 3). ATM Interface Configuration

2 Configure the adjacent ATM switch

8260 port configuration

```
8260ATM1> set port 14.1 enable ilmi_forced_sig_3_1
Port set
8260ATM1> show port 14.1 verbose

     Type  Mode     Status
-------------------------------------------------------------------------------
14.01:UNI enabled  UP-OKAY

Signalling Version   : with ILMI, forced 3.1
Flow Control         : On
Frame format         : SDH STM-1
Connector            : SC DUPLEX
Media                : Multimode fiber
Port speed           : 155000 Kbps
Remote device is active
IX status            : IX OK
Scrambling mode      : frame and cell
Clock mode           : internal
```

Figure 206 (Part 1 of 3). 8260 Switch Configuration

Obtain network prefix (first 13 bytes of ATM address)

```
8260ATM1> show device
8260 ATM Control Point and Switch Module
Name : 8260A
Location :
ITSO Raleigh, B678

For assistance contact :
Jaap de Goede

Manufacture id: VIME
Part Number: 58G9605 EC Level: C38846
Boot EEPROM version: v.1.2.0
Flash EEPROM version: v.2.0.4
Flash EEPROM backup version: v.2.0.4
Last Restart : 12:37:21 Fri 20 Sep 96 (Restart Count: 66)

A-CPSW
--------------------------------------------------------------------------------
 ATM address: 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.82.60.A1.00
```

Figure 206 (Part 2 of 3). 8260 Switch Configuration

Set the LECS well-known address (ILMI MIB entry)

```
8260ATM1> set lan_emul configuration_server
Enter WKA activity: inactive_wka
Enter ATM address : 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00
Entry set.
8260ATM1> show lan_emul configuration_server
Index            ATM address
-------------------------------------------------------------------------
 1            39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00
8260ATM1>
```

Figure 206 (Part 3 of 3). 8260 Switch Configuration

Important

The definition of the LECS address is irrelevant for this scenario but has been added for the scenarios which require LECS functions.

3 Restart 8210 and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 207 (Part 1 of 2). Restart and Verify

Display Interface Status

```
*talk 5

CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list addresses

                         ATM Address
          Network Prefix                     ESI         SEL
------------------------------------ ----------------- --
  No Addresses Registered
ATM Interface+list vccs

 Conn    Conn                         Frames       Frames     Bytes        Bytes

 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R′cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  -----
      2  ILMI    0    16           0          51           50         2623    3231
ATM Interface+
```

Figure 207 (Part 2 of 2). Restart and Verify

11.4.3 IBM 8210 Token-Ring LE Client

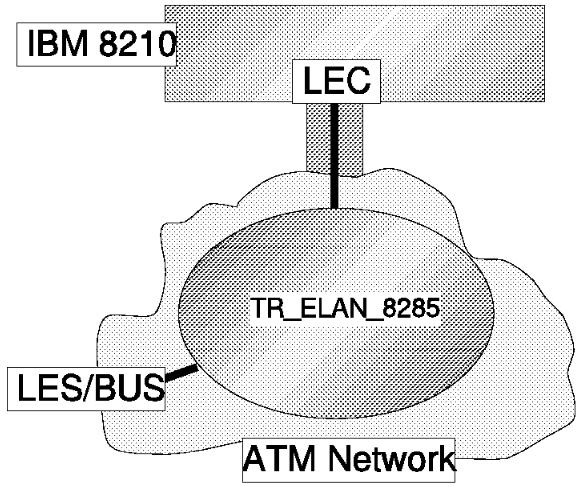
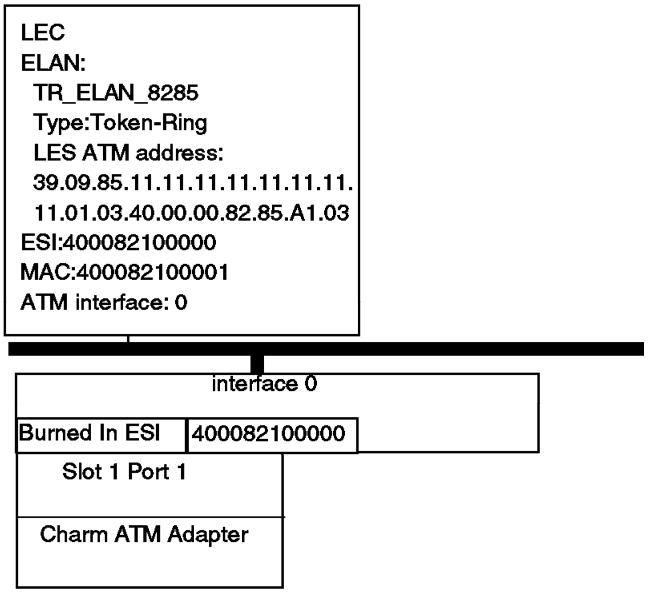


Figure 208. Token-Ring LE Client

Figure 208 depicts a configuration where the IBM 8210 Nways MSS Server contains a single LE client that connects to a token-ring ELAN that is controlled by an external LES/BUS. No LECS is used. This scenario is typically used when the customer receives the IBM 8210 Nways MSS Server and configures it for connectivity to an existing ELAN. Because of the external LES/BUS, in our case running on an IBM 8285 Nways ATM Workgroup Switch, no LES/BUS definitions are required on the MSS Server.



LEC
ELAN:
TR_ELAN_8285
Type:Token-Ring
LES ATM address:
39.09.85.11.11.11.11.11.11.11.
11.01.03.40.00.00.82.85.A1.03
ESI:400082100000
MAC:400082100001
ATM interface: 0

Burned In ESI: 400082100000

Figure 209. Token-Ring LE Client - Parameter Overview

To realize the scenario depicted in Figure 208 we used the parameters depicted in Figure 209. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 210. Configure ATM Interface

2 Define LE client

Add a token-ring LEC as an (logical) interface

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add token-ring                     1
Added Emulated LAN as interface 1
LE Client config>
```

Figure 211 (Part 1 of 3). Token-Ring LEC Definition

Configure the (logical) interface

```
LE Client config>config
Emulated LAN interface number [1]? 1  1
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8285
Token Ring Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Token Ring Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Token Ring Forum Compliant LEC Config>
```

Note:

1 Make sure the logical interface numbers are identical

Figure 211 (Part 2 of 3). Token-Ring LEC Definition

Set the LES/BUS ATM address for the LEC

```
Token Ring Forum Compliant LEC Config>set les
LES ATM address in 00.00.00.00.00.00:... form []?
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.00.82.85.A1.03
Token Ring Forum Compliant LEC Config>set auto no        2
Token Ring Forum Compliant LEC Config>list

                 ATM LEC Configuration

   ATM interface number             = 0
   LEC interface number             = 1
   LECS auto configuration          = No

   C1: Primary ATM address
          ESI address               = 40.00.82.10.00.00
          Selector byte             = 0x2
   C2: Emulated LAN type            = Token Ring
   C3: Maximum frame size           = 4544
   C5: Emulated LAN name            = TR_ELAN_8285
   C6: LE Client MAC address        = 40.00.82.10.00.01
   C7: Control timeout              = 120
   C9: LE Server ATM address        = 39.09.85.11.11.11.11.11.11.11.11.01.03.40.
00.00.82.85.A1.03
  C10: Maximum unknown count        = 1
  C11: Maximum unknown time         = 1
  C12: VCC timeout period           = 1200
  C13: Maximum retry count          = 1
  C17: Aging time                   = 300
  C18: Forward delay time           = 15
  C20: LE ARP response time         = 1
  C21: Flush timeout                = 4
  C22: Path switch delay            = 6
  C24: Multicast send VCC type      = Best-Effort
  C25: Multicast send VCC avg rate  = 155000
  C26: Multicast send VCC peak rate = 155000
  C28: Connection completion timer  = 4

   LE ARP queue depth               = 5
   LE ARP cache size                = 10
   Best effort peak rate            = 155000
   Maximum config retries           = 3
   Packet trace                     = No
   RIF Aging Timer                  = 120
   Source Routing                   = Enabled
   IPX interface configuration record missing

Token Ring Forum Compliant LEC Config>
Token Ring Forum Compliant LEC Config>exit
LE Client config>
Token Ring Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Note:

2 No autodetection of LECS

Figure 211 (Part 3 of 3). Token-Ring LEC Definition

3 Restart 8210 and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 212 (Part 1 of 2). Restart and Verify

Display Interface Status

```
*talk 5

CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
320839F4    32083A00  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02


                   *******  ADDRESSES *******

                         ATM Address
          Network Prefix                     ESI         SEL
--------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02


                   *******  VCCS *******

 Conn    Conn                         Frames       Frames     Bytes        Bytes

 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R'cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  -------
     10   P-P   0   351     32083A00            0          0            0     0
      9   P-P   0   350     32083A00            0          0            0     0
      8   P-P   0   349     32083A00            0          0            0     0
      7   P-P   0   348     32083A00            2          2          216   216
      1  SAAL   0     5            0           34         34          744  1268
      2  ILMI   0    16            0           22         21         1135  1228

ATM Interface+
```

Figure 212 (Part 2 of 2). Restart and Verify

11.4.4 IBM 8210 Ethernet LE Client

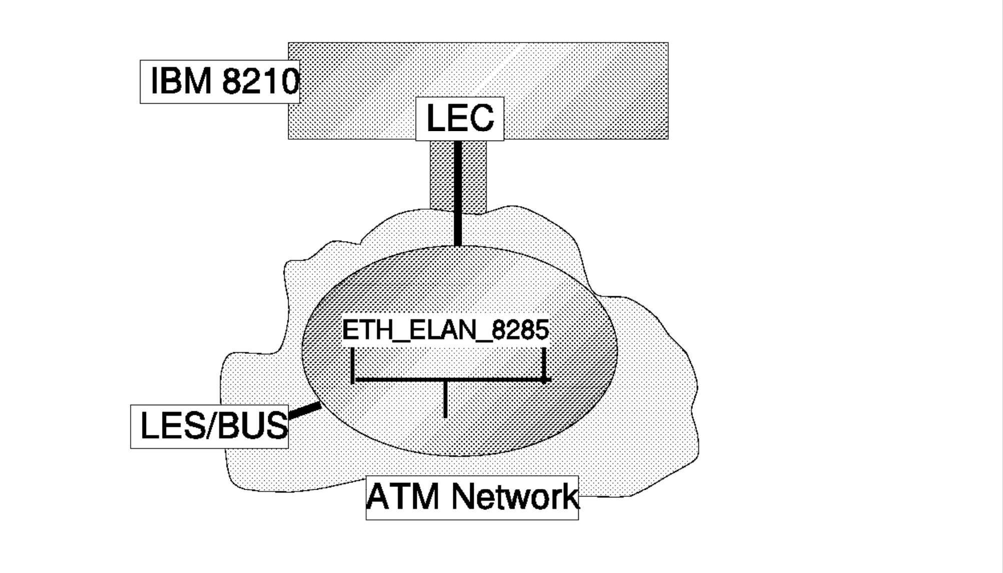
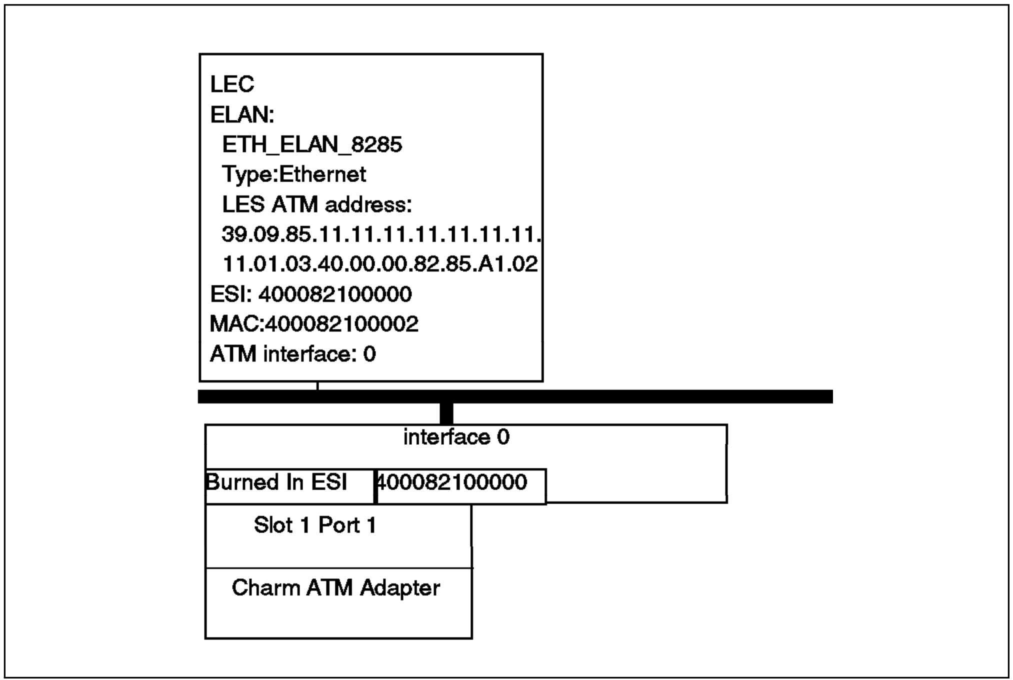


Figure 213. Ethernet LE Client

Figure 213 depicts a configuration where the IBM 8210 Nways MSS Server contains a single LE client that connects to an Ethernet ELAN that is controlled by an external LES/BUS. No LECS is used. This scenario is typically used when the customer receives the IBM 8210 Nways MSS Server and configures it for connectivity to an existing ELAN. Because of the external LES/BUS, in our case running on an IBM 8285 Nways ATM Workgroup Switch, no LES/BUS definitions are required on the MSS Server.



LEC
ELAN:
  ETH_ELAN_8285
  Type:Ethernet
  LES ATM address:
  39.09.85.11.11.11.11.11.11.11.
  11.01.03.40.00.00.82.85.A1.02
ESI: 400082100000
MAC:400082100002
ATM interface: 0

Figure 214. Ethernet LE Client - Parameter Overview

To realize the scenario depicted in Figure 213 we used the parameters depicted in Figure 214. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 215. Configure ATM Interface

2 Configure Ethernet LE client

Add an Ethernet LEC as a (logical) interface

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1                            1
LE Client config>
```

Figure 216 (Part 1 of 3). Ethernet LEC Definition

Configure the (logical) interface

```
LE Client config>config
Emulated LAN interface number [1]? 1    1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8285
Ethernet Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.02
Ethernet Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Ethernet Forum Compliant LEC Config>
```

Note:

1 Make sure the logical interface numbers are identical.

Figure 216 (Part 2 of 3). Ethernet LEC Definition

Set the LES/BUS ATM address for the LEC

```
Ethernet Forum Compliant LEC Config>set les
LES ATM address in 00.00.00.00.00.00:... form []?
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.00.82.85.A1.02
Ethernet Forum Compliant LEC Config>set auto no      2
Ethernet Forum Compliant LEC Config>list

                 ATM LEC Configuration

   ATM interface number             = 0
   LEC interface number             = 2
   LECS auto configuration          = No

   C1: Primary ATM address
          ESI address               = 40.00.82.10.00.00
          Selector byte             = 0x2
   C2: Emulated LAN type            = Ethernet
   C3: Maximum frame size           = 1516
   C5: Emulated LAN name            = ETH_ELAN_8285
   C6: LE Client MAC address        = 40.00.82.10.00.02
   C7: Control timeout              = 120
   C9: LE Server ATM address        = 39.09.85.11.11.11.11.11.11.11.11.01.03.40.
00.00.82.85.A1.02
  C10: Maximum unknown count        = 1
  C11: Maximum unknown time         = 1
  C12: VCC timeout period           = 1200
  C13: Maximum retry count          = 1
  C17: Aging time                   = 300
  C18: Forward delay time           = 15
  C20: LE ARP response time         = 1
  C21: Flush timeout                = 4
  C22: Path switch delay            = 6
  C24: Multicast send VCC type      = Best-Effort
  C25: Multicast send VCC avg rate  = 155000
  C26: Multicast send VCC peak rate = 155000
  C28: Connection completion timer  = 4

   LE ARP queue depth               = 5
   LE ARP cache size                = 10
   Best effort peak rate            = 155000
   Maximum config retries           = 3
   Packet trace                     = No
   No IPX interface configuration
   IP Encapsulation                 = ETHER
Ethernet Forum Compliant LEC Config>
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Note:

2 No autodetection of LECS

Figure 216 (Part 3 of 3). Ethernet LEC Definition

3 Restart 8210 and verify the configuration

See 3 on page 305.

11.4.5 IBM 8210 Token-Ring LE Client with LECS

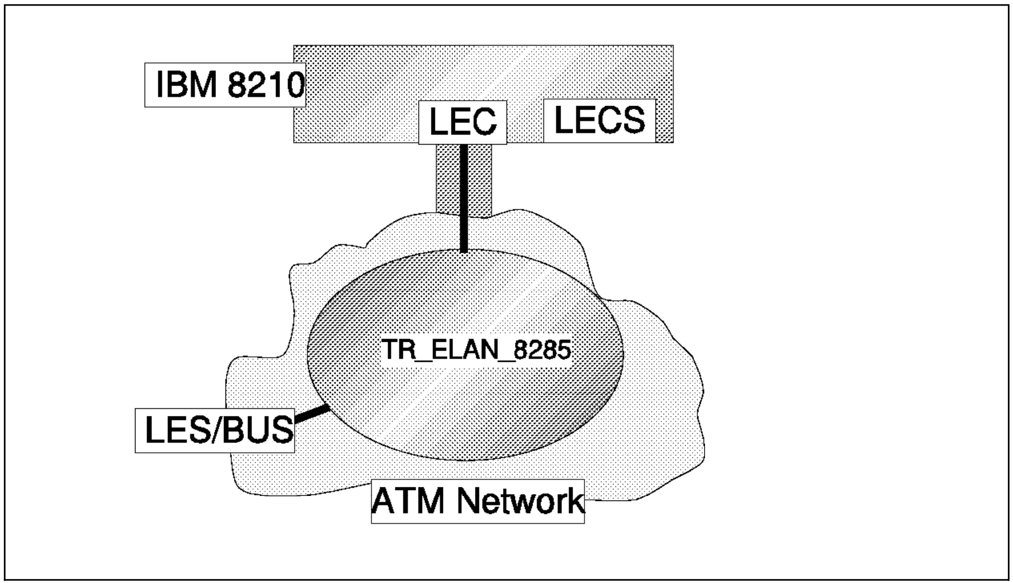


Figure 217. Token-ring LE Client and LECS

Figure 217 depicts a configuration where the IBM 8210 Nways MSS Server contains a single LE client connecting to a token-ring ELAN that is controlled by an external LES/BUS. To learn the LES/BUS address dynamically, a local LECS will be defined. The LECS assigns LE clients to the ELAN using a name policy.

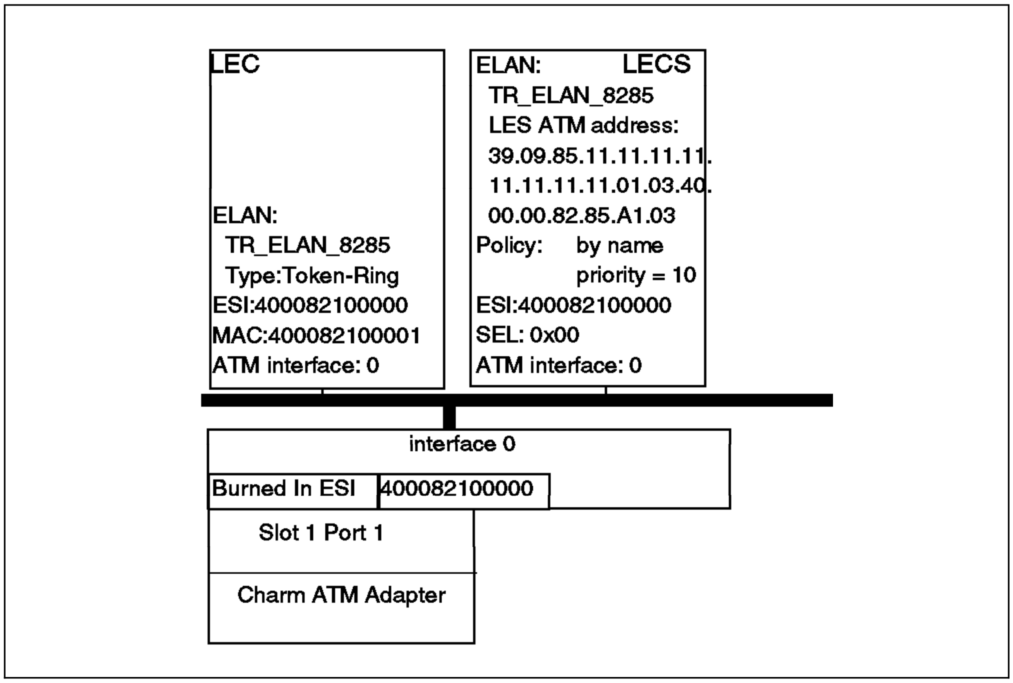


Figure 218. Token-Ring LE Client and LECS - Parameter Overview

To realize the scenario depicted in Figure 217 we used the parameters depicted in Figure 218. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 219. Configure ATM Interface

> **Important**
>
> Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 Configure the LECS

Add the LECS to the local administered ESI

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 220 (Part 1 of 4). LECS Definition

Add a token-ring ELAN definition to the LECS

```
LECS config>elan
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8285
type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN 'TR_ELAN_8285' added
Selection "ELAN addition" Complete
LECS ELANs config>
```

Figure 220 (Part 2 of 4). LECS Definition

Configure the ELAN definition, and set policy value

```
LECS ELANs config>select
  (  1) TR_ELAN_8285
Choice of ELAN [1]? 1
ELAN 'TR_ELAN_8285' selected for detailed configuration
Selected ELAN 'TR_ELAN_8285'>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 2
If primary LES is remote, enter ATM address
         []? 39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.00.82.85.A1.03
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is     []? 1
LES ATM address 39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.00.82.85.A1.03
    added to ELAN 'TR_ELAN_8285'
Selected ELAN 'TR_ELAN_8285'>policy add name
ATM address of LES for policy value(s)

        (1) 39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.00.82.85.A1.03

Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8285
ELAN name 'TR_ELAN_8285'
     bound to LES 39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.00.82.85.A1.03
Selection "ELAN name add" Complete
Selected ELAN 'TR_ELAN_8285'>exit
LECS ELANs config>exit
LECS config>
```

Figure 220 (Part 3 of 4). LECS Definition

Define general policies priority

```
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy 'byElanNm ' at priority 10
Selection "Add assignment policy" Complete
LECS POLICIES config>exit
LECS config>exit
LE Services config>exit
ATM Config>exit
Config>
```

Figure 220 (Part 4 of 4). LECS Definition

3 Configure LE client

Add a token-ring LEC as a (logical) interface

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add token-ring
Added Emulated LAN as interface 1
LE Client config>
```

Figure 221 (Part 1 of 2). Token-Ring LEC Definition

Configure the (logical) interface

```
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8285
Token Ring Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Token Ring Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Token Ring Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 221 (Part 2 of 2). Token-Ring LEC Definition

4 Restart 8210 and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 222 (Part 1 of 2). Restart and Verify

Display Interface Status

```
*talk 5


CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
32083C94    32083CA0  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02
3208373C    32083748  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00


                   *******  ADDRESSES *******

                            ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00


                   *******  VCCS *******

 Conn    Conn                          Frames       Frames     Bytes        Bytes

 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R'cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  -------
     15   P-P    0   128    32083CA0            0          0            0    0
     14   P-P    0   127    32083CA0            0          0            0    0
     13   P-P    0   126    32083CA0            0          0            0    0
     12   P-P    0   125    32083CA0            2          2          216    216
      1  SAAL    0     5           0           34         34         1072    1324
      2  ILMI    0    16           0           22         21         1142    1234
ATM Interface+
```

Figure 222 (Part 2 of 2). Restart and Verify

11.4.6 IBM 8210 Token-Ring LE Client, LES/BUS and LECS

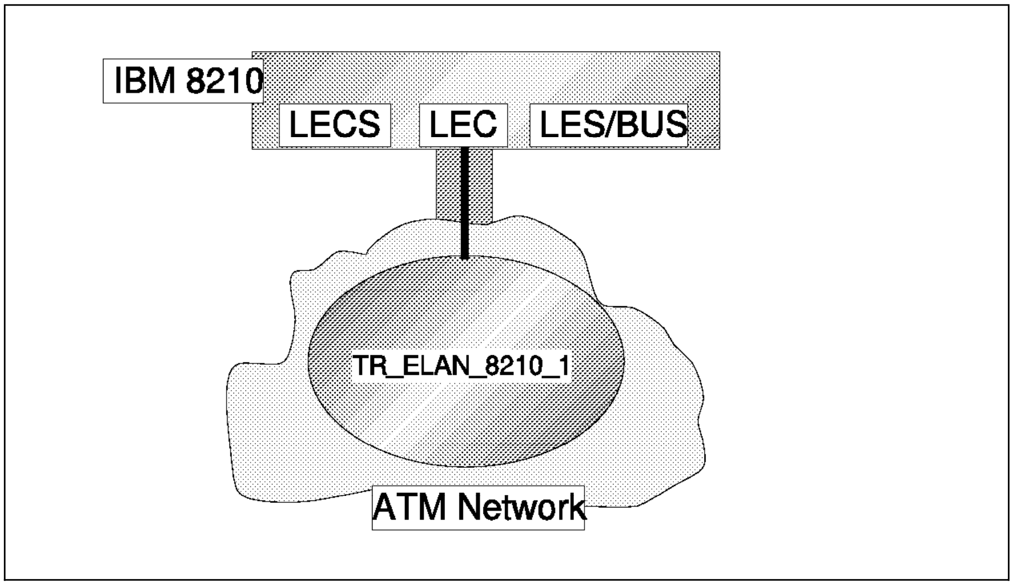


Figure 223. Token-Ring LE Client, LES/BUS, and LECS

Figure 223 depicts a configuration where the IBM 8210 Nways MSS Server contains a single LE client connecting to a token-ring ELAN that is controlled by an internal LES/BUS. For this ELAN we have activated the BCM and SRM functions for IP, IPX, and NetBIOS. To learn the LES/BUS address dynamically, a local LECS will be defined. No security will be enforced and external LE clients can join the ELAN by learning the address from the LECS or by directly addressing the LES ATM address. The LECS assigns LE clients to the ELAN using a name policy.

Note: To enable IP access to the MSS Server′s management functions we have added an IP address to the LEC. This IP address is not relevant for the LEC, LECS, and LES/BUS functions

To realize the scenario depicted in Figure 223 we used the parameters depicted in Figure 224 on page 317. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 225. Configure ATM Interface

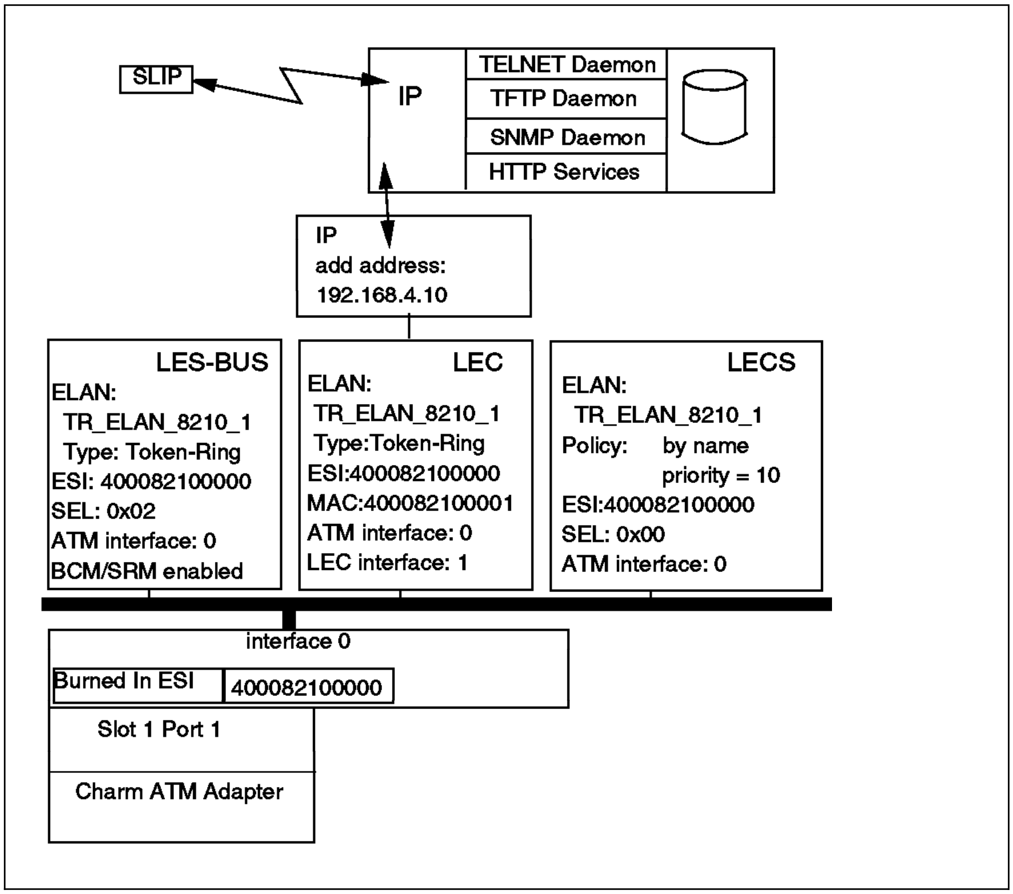


Figure 224. LECS, Token-Ring LEC, and LES/BUS - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 Define the LECS

Add the LECS to the locallly administered ESI

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:
Standard ELS activated for LECS
LECS config>
```

Figure 226 (Part 1 of 4). LECS Definition

Add a token-ring ELAN definition to the LECS

```
LECS config>elan
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 226 (Part 2 of 4). LECS Definition

Configure the ELAN definition, and set policy value

```
LECS ELANs config>select
  (  1) TR_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN ′TR_ELAN_8201_1′ selected for detailed configuration
Selected ELAN ′TR_ELAN_8210_1′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_1
    added to ELAN ′TR_ELAN_8210_1′
Selected ELAN ′TR_ELAN_8210_1′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1
ELAN name ′TR_ELAN_8210_1′
     bound to LES Local LES for: TR_ELAN_8210_1
Selection ″ELAN name add″ Complete
Selected ELAN ′TR_ELAN_8210_1′>exit
LECS ELANs config>exit
LECS config>
```

Figure 226 (Part 3 of 4). LECS Definition

Define general policies priority

```
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete
LECS POLICIES config>exit
LECS Config>exit
LE Services config>exit
ATM Config>exit
Config>
```

Figure 226 (Part 4 of 4). LECS Definition

3 **Define the token-ring LES/BUS**

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN 'TR_ELAN_8210_1'>add
Turn on Standard Event Logging for LES [yes]
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'TR_ELAN_8210_1'>enable bcm all   1
LES-BUS config for ELAN 'TR_ELAN_8210_1'>enable source    2
LES-BUS config for ELAN 'TR_ELAN_8210_1'>exit
LE Services config>exit
ATM Config>exit
Config>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

2 Enabling Source Routing Management for NetBIOS, IP and IPX

Figure 227. Set Up Token-Ring LES/BUS Pair

4 Define the token-ring LEC

Add a token-ring LEC as an (logical) interface

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add token-ring
Added Emulated LAN as interface 1
LE Client config>
```

Figure 228 (Part 1 of 2). Token-Ring LEC Definition

Configure the (logical) interface

```
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_1
Token Ring Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Token Ring Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x3
Token Ring Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 228 (Part 2 of 2). Token-Ring LEC Definition

5 Assign IP address to the LEC (optional)

```
*talk 6

Config>protocol ip
Internet protocol user configuration
IP config>add address 1                                 1
New address [0.0.0.0]? 192.168.4.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Note:

1 Make sure you use the logical interface of the token-ring LE client

Figure 229. Add IP to the LEC

6 Restart and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 230 (Part 1 of 4). Restart and Verify

Display LES operation

```
*talk 5

+network 0
ATM Console
ATM+le-services
LE-Services Console
LE-SERVICES+work
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LE-Services Console for an existing LES-BUS Pair
EXISTING LES-BUS ′TR_ELAN_8210_1′+database list all lec
Number of LEC′s to display: 1

    LEC-LES and LEC-BUS State  (UP=Up,  ID=Idle,  --. --.
      **=Other; Show specific LEC to see actual)    v   v
                                             LEC    State   #ATM  #Reg   #Lrnd
LEC Primary ATM Address               Proxy  ID    LES BUS  Adrs  MACs    MACs
----------------------------------------  -  ----   --  --  ----  -----  -----
3909851111111111111111010140008210000003  N  0001   UP  UP     1      1      0
EXISTING LES-BUS ′TR_ELAN_8210_1′+
```

Figure 230 (Part 2 of 4). Restart and Verify

Display interface status

```
+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
32083E88    32083E94  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.03
32083930    3208393C  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00
32083768    32083774  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02

                   *******  ADDRESSES *******

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.03
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00


                   *******  VCCS *******

 Conn    Conn                        Frames       Frames     Bytes        Bytes

 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R'cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  -------
     18  P-MP    0   381    32083774            0          0            0    0
     19   P-P    0   382    32083E94            0          0            0    0
     16   P-P    0   379    32083E94            0          0            0    0
     17   P-P    0   380    32083774            0          0            0    0
     14  P-MP    0   377    32083774            0          0            0    0
     15   P-P    0   378    32083E94            0          0            0    0
     12   P-P    0   375    32083E94            2          2          216    216
     13   P-P    0   376    32083774            0          2            0    216
      1  SAAL    0     5           0          241        241         4364   8908
      2  ILMI    0    16           0           65         65         3456   4075
ATM Interface+
```

Figure 230 (Part 3 of 4). Restart and Verify

Verify ESI registration (on ATM switch)

```
8260ATM1> show atm_esi
Enter module: 14.
Enter port: 1
Port   ATM_ESI           Type
-------------------------------------------------------------------------
14.01 40.00.82.10.00.00 dynamic
8260ATM1>
```

Figure 230 (Part 4 of 4). Restart and Verify

11.4.7 IBM 8210 Ethernet LE Client, LES/BUS, and LECS

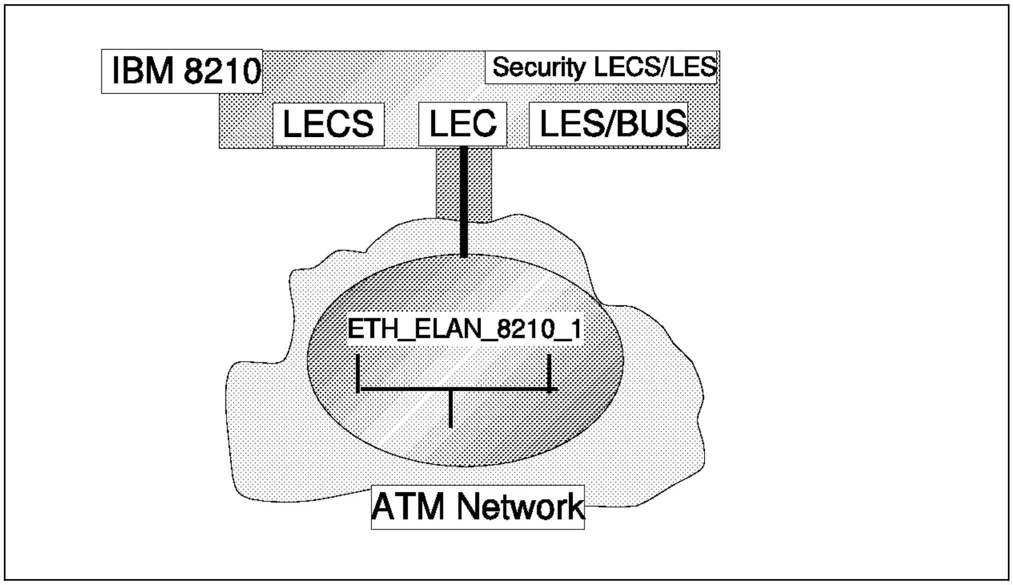


Figure 231. Ethernet LE Client, Internal LES/BUS, and External LECS

Figure 231 depicts a configuration where the IBM 8210 Nways MSS Server contains a single LE client connecting to an Ethernet ELAN that is controlled by an internal LES/BUS. To learn the LES/BUS address dynamically, a local LECS will be defined. The LECS assigns LE clients to the ELAN using a name policy. To prevent LE clients from joining the ELAN without proper configuration the LECS/LES security interface has been enabled, and the ELAN defined as a secure ELAN.

For this ELAN we have activated BCM and SRM functions for IP, IPX, and NetBIOS.

Note: To enable IP access to the MSS Server′s management functions we have added an IP address to the LE client. This IP address is not relevant for the LEC, LECS, and LES/BUS functions

To realize the scenario depicted in Figure 231 we used the parameters depicted in Figure 232 on page 325. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 233. Configure ATM Interface

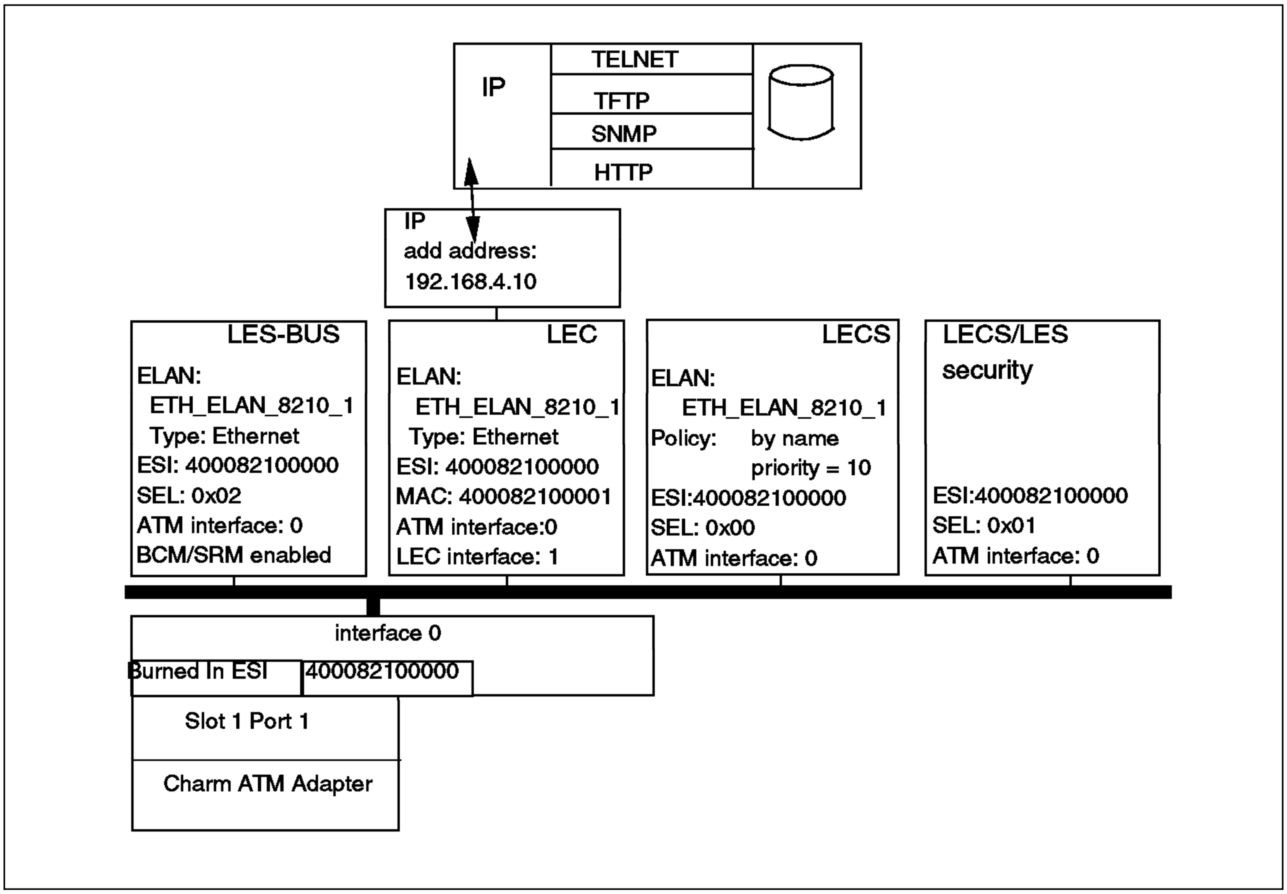


Figure 232. LECS, Ethernet LEC, and LES/BUS - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 Configure the LECS

Add the LECS to the user-defined ESI

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:
Standard ELS activated for LECS
LECS config>
```

Figure 234 (Part 1 of 4). LECS Definition

Add an Ethernet ELAN definition to the LECS

```
LECS config>elan
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? ETH_ELAN_8210_1
type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 1
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 1
ELAN ’ETH_ELAN_8210_1’ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 234 (Part 2 of 4). LECS Definition

Configure the ELAN definition, and set policy values

```
LECS ELANs config>select
  (  1) ETH_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN ’ETH_ELAN_8210_1’ selected for detailed configuration
Selected ELAN ’ETH_ELAN_8210_1’>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: ETH_ELAN_8210_1
    added to ELAN ’ETH_ELAN_8210_1’
Selected ELAN ’ETH_ELAN_8210_1’>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: ETH_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? ETH_ELAN_8210_1
ELAN name ’ETH_ELAN_8210_1’
     bound to LES Local LES for: ETH_ELAN_8210_1
Selection ″ELAN name add″ Complete
Selected ELAN ’ETH_ELAN_8210_1’>exit
LECS ELANs config>exit
LECS config>
```

Figure 234 (Part 3 of 4). LECS Definition

```
Define general policies priority
```

```
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete
LECS POLICIES config>exit
LECS config>exit
```

Figure 234 (Part 4 of 4). LECS Definition

3 Configure LECS/LES security interface

```
LE Services config>security
LECS Interface configuration
LECS INTERFACE config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
Select ESI [1]? 2
Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.
LECS Interface Selector (in hex) [1]? 1
LECS INTERFACE config>list
LECS Interface Detailed Configuration
   LECS Interface Enabled/Disabled: Enabled
   ATM Device number: 0
   ESI: 40.00.82.10.00.00
   Selector: 0x01
   Configuration Direct VCC Traffic Type: Best Effort VCC
   Configuration Direct VCC PCR in Kbps: 155000
   Configuration Direct VCC SCR in Kbps: 0
LECS INTERFACE config>exit
LE Services config>
```

Figure 235. Configure LECS/LES Security Interface

4 Configure the Ethernet LES/BUS

Define the LES/BUS pair

```
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN ′ETH_ELAN_8210_1′>add
Turn on Standard Event Logging for LES [yes]
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 2
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′ETH_ELAN_8210_1′>enable bcm all    1
LES-BUS config for ELAN ′ETH_ELAN_8210_1′>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

Figure 236 (Part 1 of 2). Set Up an Ethernet LES/BUS Pair

Activate security for the selected ELAN

```
LES-BUS config for ELAN ′ETH_8210_1′>set security
Enable Security (LECS validation of Joins) [no]yes
Selection ″Enable Security (LECS Validation of Joins)″ Complete
LES-BUS config for ELAN ′ETH_8210_1′>exit
LE Services config>exit
ATM Config>
```

Figure 236 (Part 2 of 2). Set Up an Ethernet LES/BUS Pair

5 Configure an Ethernet LEC

Add an Ethernet LEC as a (logical) interface

```
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1
LE Client config>
```

Figure 237 (Part 1 of 2). Ethernet LEC Definition

Configure the (logical) interface

```
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8210_1
Ethernet Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Ethernet Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x3
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 237 (Part 2 of 2). Ethernet LEC Definition

6 Assign IP address to the LEC (optional)

```
Config>protocol ip
Internet protocol user configuration
IP config>add address 1                                    1
New address [0.0.0.0]? 192.168.4.10
Address mask [255.255.255.0]?
IP config>
```

Note:

1 Make sure you use the logical interface of the Ethernet LE client

Figure 238. Adding IP to an LEC Interface

7 Restart and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

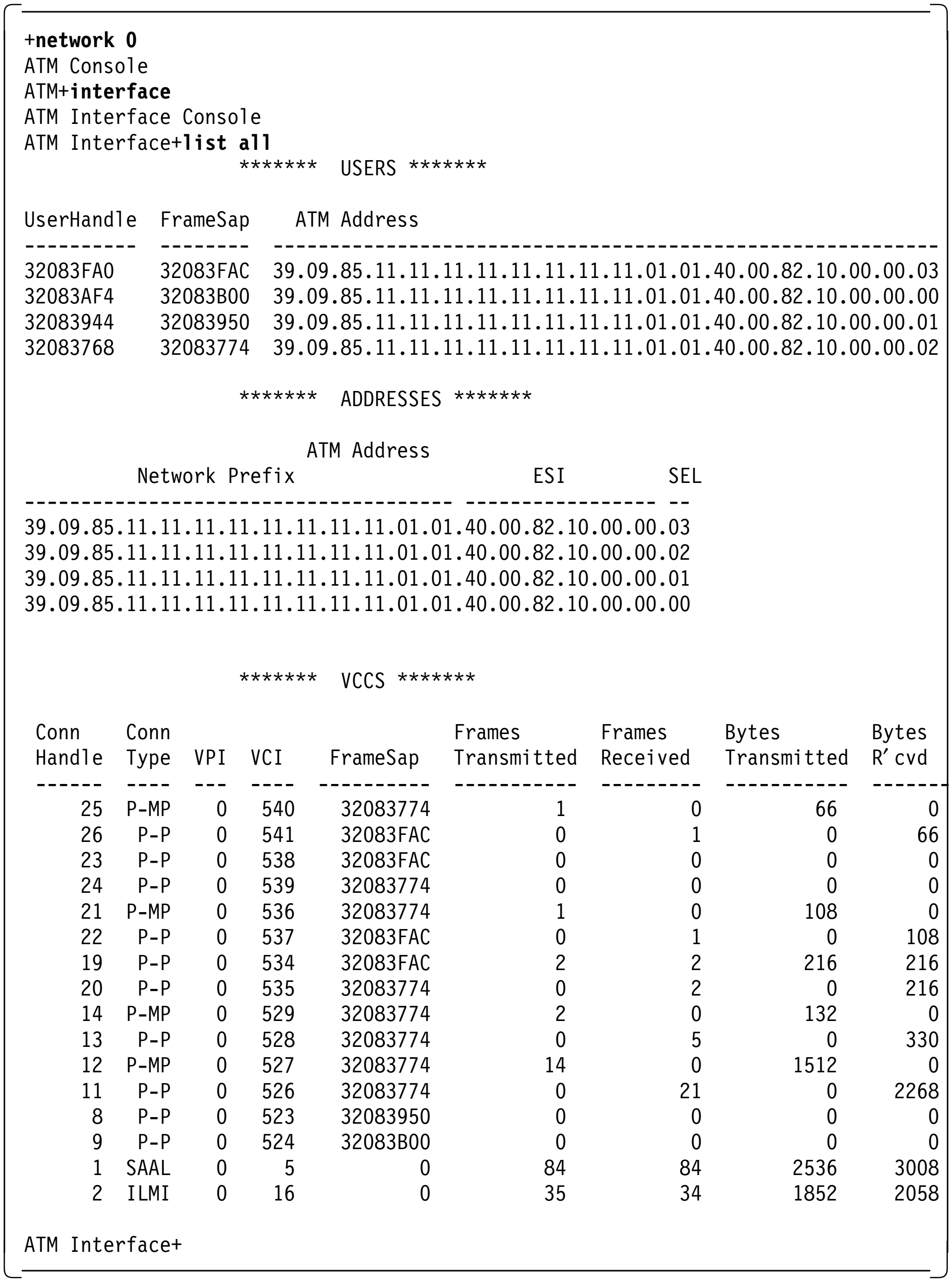
Figure 239 (Part 1 of 4). Restart and Verify

Verify LES operation

```
*talk 5

+network 0
ATM Console
ATM+le-services
LE-Services Console
LE-SERVICES+work
ELAN Name (ELANxx) []? ETH_ELAN_8210_1
LE-Services Console for an existing LES-BUS Pair
EXISTING LES-BUS ′ETH_ELAN_8210_1′+database list all lec
Number of LEC′s to display: 2

    LEC-LES and LEC-BUS State  (UP=Up,  ID=Idle,  --. --.
      **=Other; Show specific LEC to see actual)    v   v
                                             LEC    State   #ATM  #Reg   #Lrnd
LEC Primary ATM Address               Proxy  ID    LES BUS  Adrs  MACs    MACs
----------------------------------------  -  ----   --  --  ----  -----  -----
3909851111111111111111010140008281000000  Y  0001   UP  UP     1      5      0
3909851111111111111111010140008210000003  N  0002   UP  UP     1      1      0
EXISTING LES-BUS ′ETH_ELAN_8210_1′+
```

Figure 239 (Part 2 of 4). Restart and Verify

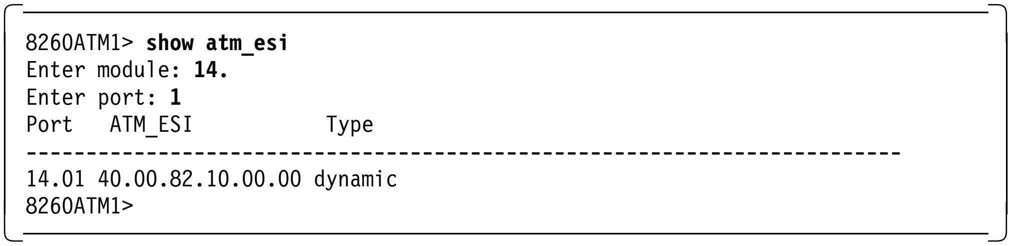
Display interface status



```
+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
32083FA0    32083FAC  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.03
32083AF4    32083B00  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00
32083944    32083950  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.01
32083768    32083774  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02

                   *******  ADDRESSES *******

                         ATM Address
          Network Prefix                     ESI         SEL
--------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.03
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.01
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.00


                   *******  VCCS *******

 Conn    Conn                           Frames       Frames     Bytes        Bytes
 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R'cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  -------
     25  P-MP    0   540    32083774            1          0           66        0
     26   P-P    0   541    32083FAC            0          1            0       66
     23   P-P    0   538    32083FAC            0          0            0        0
     24   P-P    0   539    32083774            0          0            0        0
     21  P-MP    0   536    32083774            1          0          108        0
     22   P-P    0   537    32083FAC            0          1            0      108
     19   P-P    0   534    32083FAC            2          2          216      216
     20   P-P    0   535    32083774            0          2            0      216
     14  P-MP    0   529    32083774            2          0          132        0
     13   P-P    0   528    32083774            0          5            0      330
     12  P-MP    0   527    32083774           14          0         1512        0
     11   P-P    0   526    32083774            0         21            0     2268
      8   P-P    0   523    32083950            0          0            0        0
      9   P-P    0   524    32083B00            0          0            0        0
      1  SAAL    0     5           0           84         84         2536     3008
      2  ILMI    0    16           0           35         34         1852     2058

ATM Interface+
```

Figure 239 (Part 3 of 4). Restart and Verify

Verify ESI registration (on ATM switch)



```
8260ATM1> show atm_esi
Enter module: 14.
Enter port: 1
Port   ATM_ESI           Type
-------------------------------------------------------------------------
14.01 40.00.82.10.00.00 dynamic
8260ATM1>
```

Figure 239 (Part 4 of 4). Restart and Verify

11.4.8 IBM 8210 LIS Client

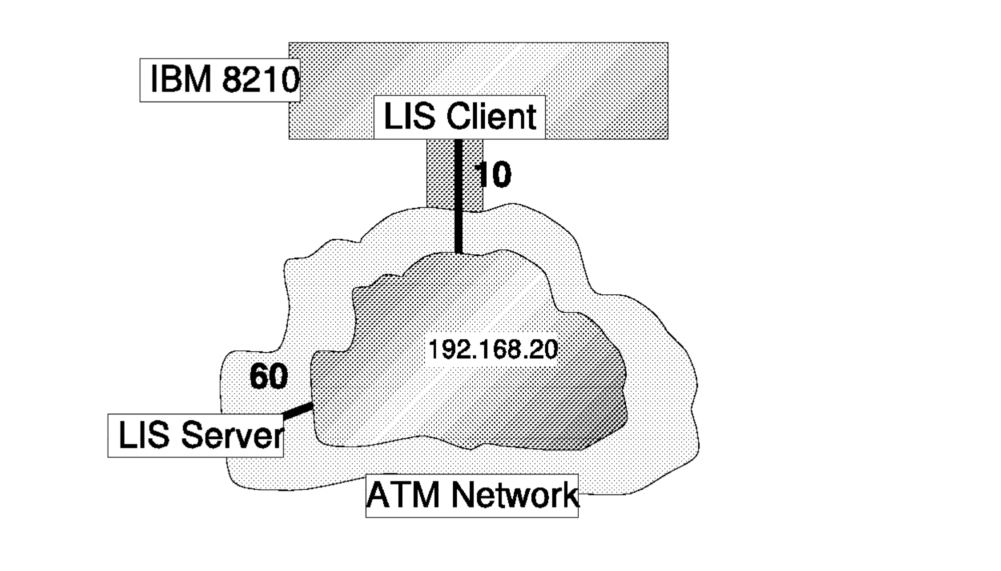


Figure 240. Defining a LIS Client

Figure 240 depicts a configuration where the IBM 8210 Nways MSS Server contains a single LIS client that connects to a Classical IP subnet (LIS). An external LIS (ARP) server is being used. This scenario is typically used when the customer receives the IBM 8210 Nways MSS Server and configures it for initial IP over ATM (*in-band*) connectivity.

Note: The LIS client′s IP address provides access to the MSS Server′s management functions.

To realize the scenario depicted in Figure 240 we used the parameters depicted in Figure 241 on page 333. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 50.00.82.10.00.00
ATM Interface Config>exit
ATM Config>exit
Config>
```

Figure 242. Configuring the ATM Interface

2 Configure the LIS Client

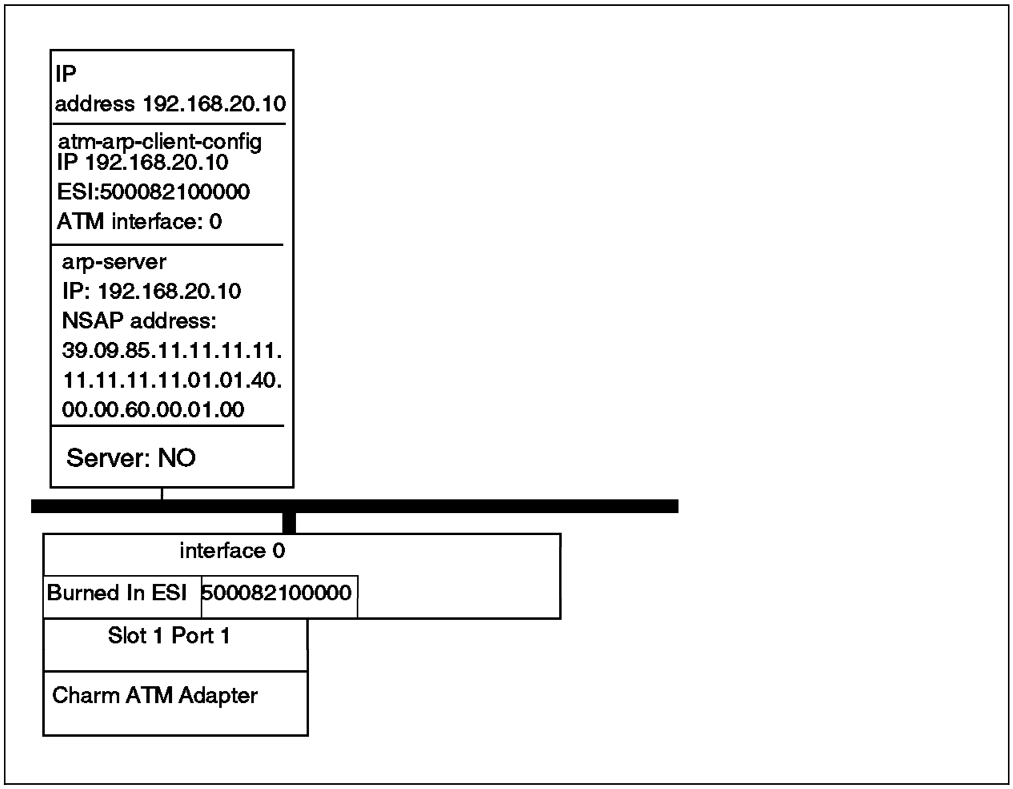


Figure 241. LIS Client - Parameter Overview

Configure IP on the ATM interface

```
Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0                1
New address [0.0.0.0]? 192.168.20.10                2
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 243 (Part 1 of 3). Setup of the LIS Client

Configure LIS client

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0                          1
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.20.10       2
This client is also a server? [No]: no
Refresh timeout (in minutes) [5]?
Enable auto-refresh? [No]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 500082100000
Select ESI [1]? 2
Use internally assigned selector? [Yes]:
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>
```

Figure 243 (Part 2 of 3). Setup of the LIS Client

Assign the ARP Server for the LIS client

```
ARP config>add arp-server private-nsapa
Local Client IP Address [0.0.0.0]? 192.168.20.10  2
Private NSAP Address: Specify 40 digits
ATM Address []? 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.60.00.01.00
ARP config>exit
Config>
```

Note:

1 Make sure the logical interface numbers match

2 Make sure the IP addresses match

Figure 243 (Part 3 of 3). Setup of the LIS Client

3 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 244 (Part 1 of 3). Restart and Verify

Verify the IP configuration

```
*talk 5

CGW Operator Console

+protocol ip
IP>ping 192.168.20.60                             1
PING 192.168.20.10 -> 192.168.20.60: 56 data bytes, ttl=64, every 1 sec.
56 data bytes from 192.168.20.60: icmp_seq=0. ttl=64. time=0. ms
56 data bytes from 192.168.20.60: icmp_seq=1. ttl=64. time=0. ms
56 data bytes from 192.168.20.60: icmp_seq=2. ttl=64. time=0. ms

----192.168.20.60 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
IP>exit
```

Note:

1 Pinging to ARP server (192.168.20.60)

Figure 244 (Part 2 of 3). Restart and Verify

Display interface status

```
+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
320B3CC8    320B3CD4  39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.C8


                   *******  ADDRESSES *******

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.C8


                   *******  VCCS *******

 Conn    Conn                         Frames       Frames     Bytes        Bytes
 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R'cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  ------
      7   P-P    0   399    320B3CD4           1           5          44      436
      1  SAAL    0     5           0          11          11         212      136
      2  ILMI    0    16           0          31          30        1578     1771

ATM Interface+
```

Figure 244 (Part 3 of 3). Restart and Verify

11.4.9 IBM 8210 ARP Server

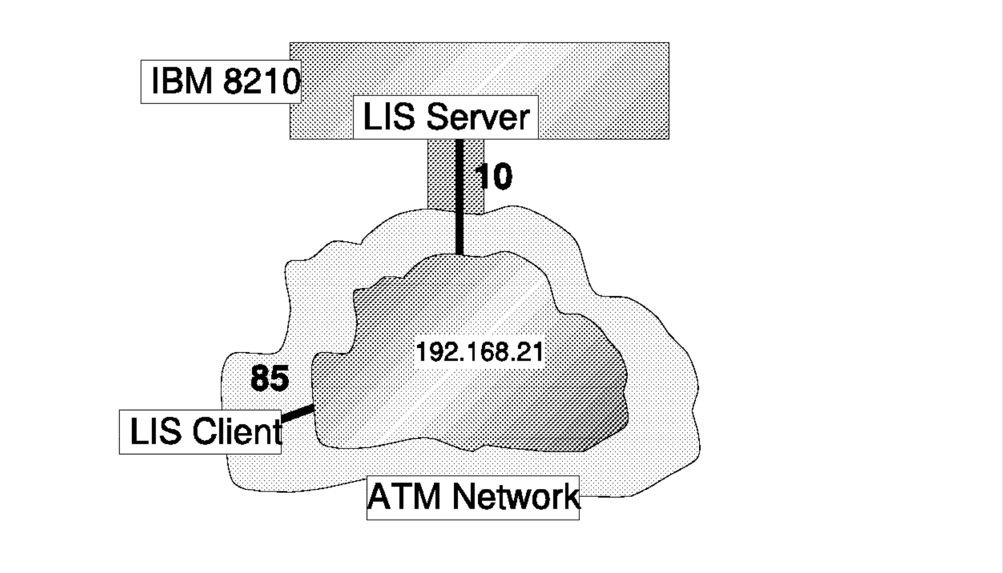


Figure 245. Defining an ARP Server

Figure 245 depicts a configuration where the IBM 8210 Nways MSS Server contains an ARP server to establish a Classical IP subnet (LIS). As the 8210 ARP server definition includes LIS client functions, IP connectivity to external LIS clients results.

This scenario is typically used when the customer receives the IBM 8210 Nways MSS Server and configures it for initial IP over ATM (*in-band*) connectivity.

Note: The ARP server's IP address provides access to the MSS Server's management functions.

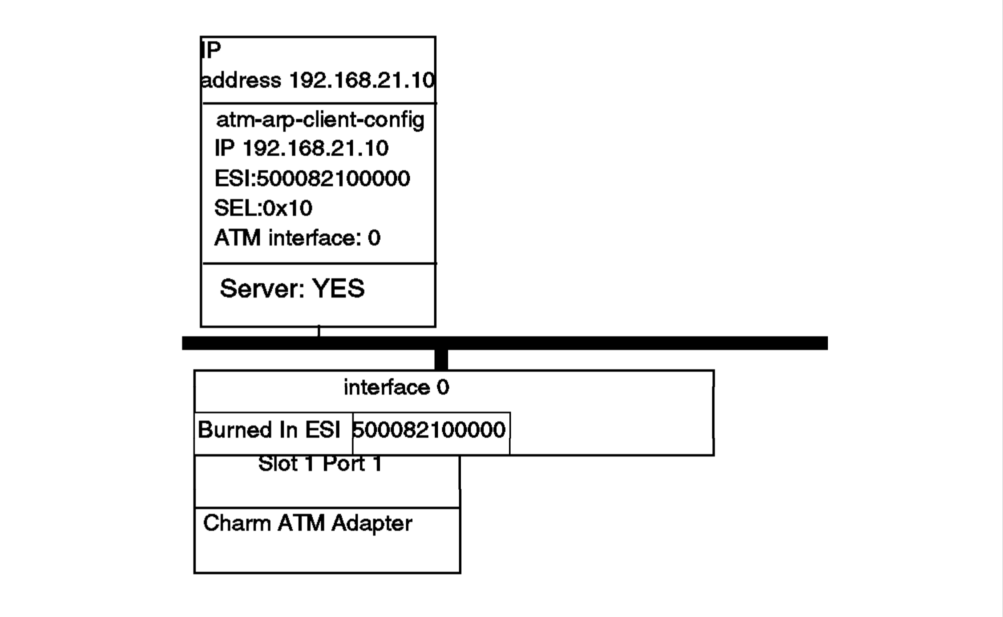


Figure 246. ARP Server - Parameter Overview

To realize the scenario depicted in Figure 245 we used the parameters depicted in Figure 246. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 50.00.82.10.00.00
ATM Interface Config>exit
ATM Config>exit
Config>
```

Figure 247. Configure ATM Interface

2 Configure ARP server

Define IP address

```
Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0                1
New address [0.0.0.0]? 192.168.21.10                2
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 248 (Part 1 of 2). Configure ARP Server

Configure the ARP server

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0                            1
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.21.10         2
This client is also a server? [No]: yes
Refresh timeout (in minutes) [20]?
Enable auto-refresh? [Yes]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 500082100000
Select ESI [1]? 2
Use internally assigned selector? [Yes]: no        3
Selector Only, Range 00..FF [00]? 10
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Note:

1 Make sure the logical interface numbers match

2 Make sure the IP addresses match

3 Use user-defined selector for ARP server

Figure 248 (Part 2 of 2). Configure ARP Server

3 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 249 (Part 1 of 3). Restart and Verify

Verify the IP configuration

```
*talk 5

CGW Operator Console

+protocol ip
IP>ping 192.168.21.85                               1
PING 192.168.21.10 -> 192.168.21.85: 56 data bytes, ttl=64, every 1 sec.
56 data bytes from 192.168.21.85: icmp_seq=0. ttl=64. time=0. ms
56 data bytes from 192.168.21.85: icmp_seq=1. ttl=64. time=0. ms
56 data bytes from 192.168.21.85: icmp_seq=2. ttl=64. time=0. ms

----192.168.21.85 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
IP>exit
```

Note:

1 Pinging to LIS client (192.168.21.85)

Figure 249 (Part 2 of 3). Restart and Verify

Display interface status

```
+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
320B3C84    320B3C90  39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.10


                   *******  ADDRESSES *******

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.10


                   *******  VCCS *******

 Conn    Conn                         Frames       Frames     Bytes        Bytes
 Handle Type   VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R’cvd
 ------ ----   ---  ----  ----------  -----------  ---------  -----------  ------
      7  P-P     0   401    320B3C90            3          3          156     296
      6  P-P     0   400    320B3C90            3          1          176     108
      1 SAAL     0     5           0          107        107         1124    1300
      2 ILMI     0    16           0          313        312        16644   20107

ATM Interface+
```

Figure 249 (Part 3 of 3). Restart and Verify

11.4.10 IBM 8210 LIS Client to LIS Client Using PVC

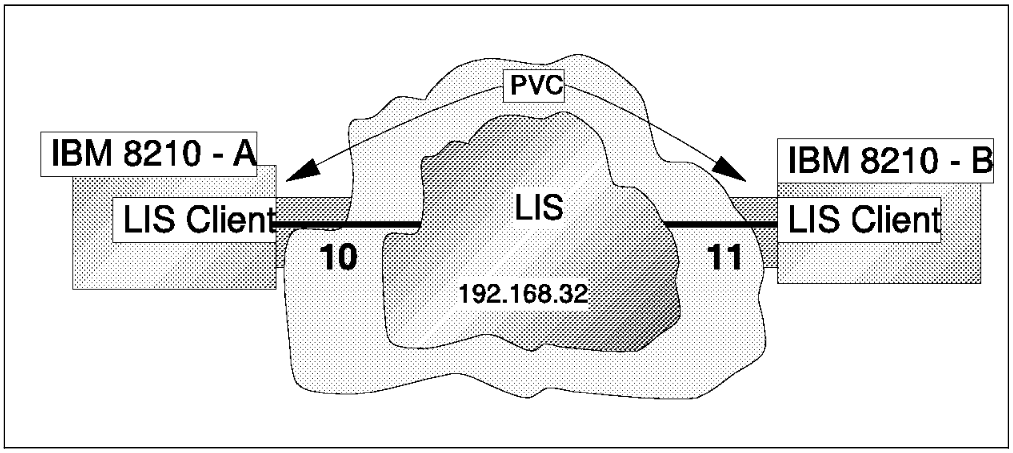


Figure 250. LIS to LIS Client Using PVC

Figure 250 depicts a configuration where an IBM 8210 Nways MSS Server contains a single LIS client that connects to another IBM 8210 Nways MSS Server. A PVC is used to connect the two LIS Clients. This scenario is typically used when the customer receives the IBM 8210 Nways MSS Server and configures it for IP using PVCs. PVCs are network controlled instead of SVCs which are endstation controlled.

To realize the scenario depicted in Figure 250, we used the parameters depicted in Figure 251 on page 341. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

When using PVC′s, it is not necessary to define any ESIs. Because we are using interface 0, which is available by default, no configuration is required other than setting the proper UNI version.

Set UNI version

```
*talk 6

Config>list devices
Ifc 0 CHARM ATM PCI Adapter          Slot: 1  Port: 1
Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>
```

Figure 252. ATM Interface Configuration

Note: This configuration is identical on both 8210s.

2 **Configure the PVC** (On ATM switch)

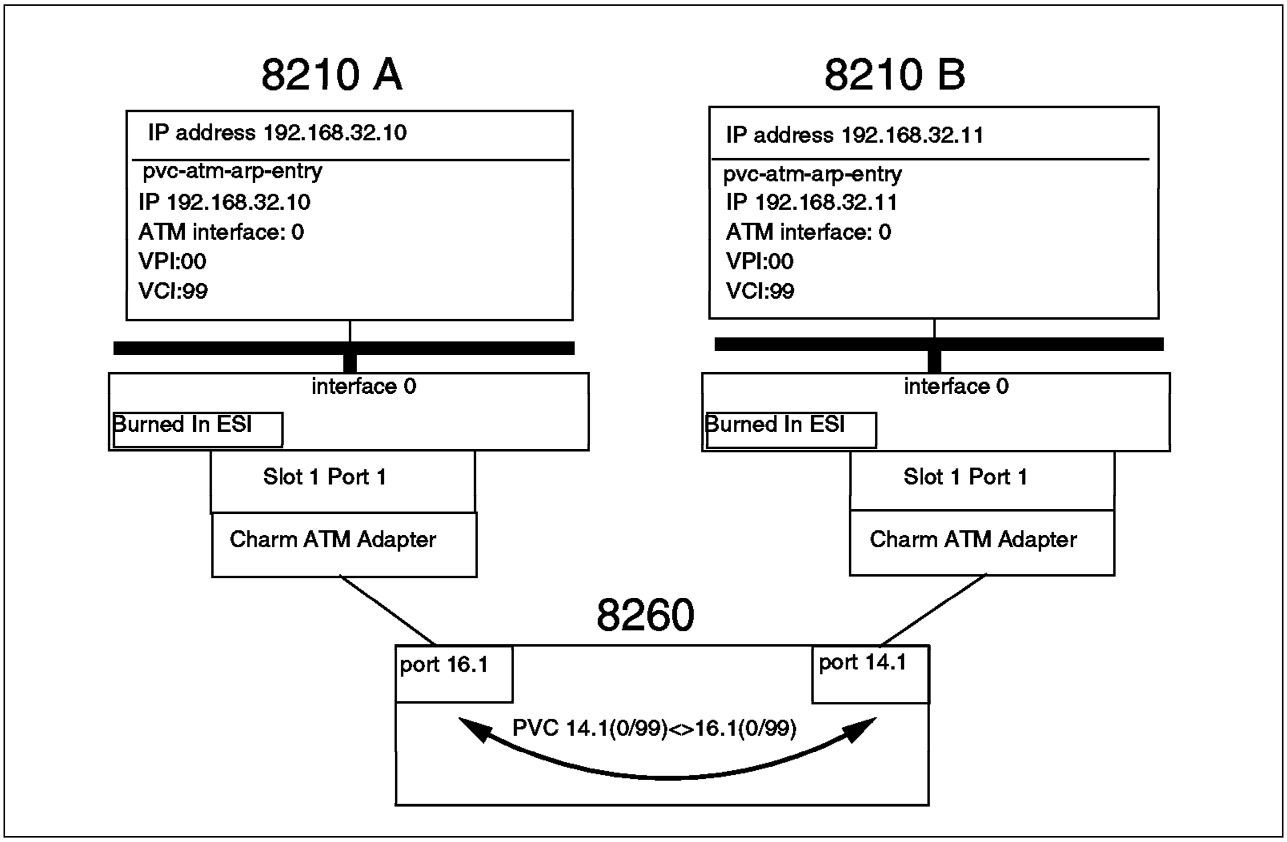


Figure 251. LIS Clients Using PVC - Parameter Overview

```
8260ATM1> set pvc
Enter local port: 14.
Enter local port: 1
Enter PVC id: 56
Enter remote port: 16.
Enter remote port: 1
Enter remote hub number: 1
Enter call type: channel
Enter local VPI: 0.
Enter local VCI: 99
Enter remote VPI: 0.
Enter remote VCI: 99
Enter quality of service: best_effort
PVC set and started.
8260ATM1> show pvc
Enter port: 14.
Enter port: 1
Enter pvc id: 56

       Local end point       ! Remote end point  !
-----------------------------+-------------------+
 Port   id   type   Vpi/Vci  ! Port Vpi/Vci   HNb!   role  !QOS! Status
-----------------------------+-------------------+---------+---+---------
14.01    56 PTP-PVC   0/99   !16.01   0/99      1! Primary ! BE!Active
8260ATM1>
```

Figure 253. Configuring 8260 PVC

3 **Configure the LIS client on 8210 A**

Define IP address

```
*talk 6

Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.32.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 254 (Part 1 of 3). Configuring 8210 A

Configure the LIS client characteristics 1

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.32.10
This client is also a server? [No]: no
Refresh timeout (in minutes) [5]?
Enable auto-refresh? [No]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
Select ESI [1]?
Use internally assigned selector? [Yes]:
Selector Only, Range 00..FF [00]?
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Note:

1 LIS client definition is optional. Implicit LIS client, with default settings, will result if no atm-arp-client-configuration is entered.

Figure 254 (Part 2 of 3). Configuring 8210 A

Configure the PVC ARP entry

```
Config>protocol arp
ARP user configuration
ARP config>add pvc-atm-arp-entry
Interface Number [0]? 0
Protocol [IP]?
Local client IP Address [0.0.0.0]? 192.168.32.10
Specify destination protocol address? [Yes]: no    1
Permanent Virtual Circuit VPI, Range 0..255 [0]? 0
Permanent Virtual Circuit VCI, Range 0..65535 [0]? 99
ARP config>exit
Config>
```

Note:

1 Partner IP address will be learned dynamically

Figure 254 (Part 3 of 3). Configuring 8210 A

4 Configure the LIS client on 8210 B

Define IP address

```
*talk 6

Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.32.11
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 255 (Part 1 of 3). Configuring 8210 B

Configure the LIS client characteristics 1

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.32.11
This client is also a server? [No]: no
Refresh timeout (in minutes) [5]?
Enable auto-refresh? [No]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
Select ESI [1]?
Use internally assigned selector? [Yes]:
Selector Only, Range 00..FF [00]?
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Note:

1 LIS client definition is optional. Implicit LIS client, with default settings, will result if no atm-arp-client-configuration is entered.

Figure 255 (Part 2 of 3). Configuring 8210 B

Configure the PVC ARP entry

```
Config>protocol arp
ARP user configuration
ARP config>add pvc-atm-arp-entry
Interface Number [0]? 0
Protocol [IP]?
Local client IP Address [0.0.0.0]? 192.168.32.11
Specify destination protocol address? [Yes]: no
Permanent Virtual Circuit VPI, Range 0..255 [0]? 0
Permanent Virtual Circuit VCI, Range 0..65535 [0]? 99
ARP config>exit
Config>
```

Figure 255 (Part 3 of 3). Configuring 8210 B

5 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 256 (Part 1 of 2). Restart and Verify

Verify the IP configuration

```
*talk 5

CGW Operator Console

+protocol ip
IP>ping 192.168.32.11                                    1
PING 192.168.32.10 -> 192.168.32.11: 56 data bytes, ttl=64, every 1 sec.
56 data bytes from 192.168.32.11: icmp_seq=0. ttl=64. time=0. ms
56 data bytes from 192.168.32.11: icmp_seq=1. ttl=64. time=0. ms
56 data bytes from 192.168.32.11: icmp_seq=2. ttl=64. time=0. ms

----192.168.32.11 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
IP>exit
```

Note:

1 Pinging to remote LIS client (192.168.32.11)

Figure 256 (Part 2 of 2). Restart and Verify

11.4.11 IBM 8210 LIS Client to LIS Client Using SVC, no ARP Server

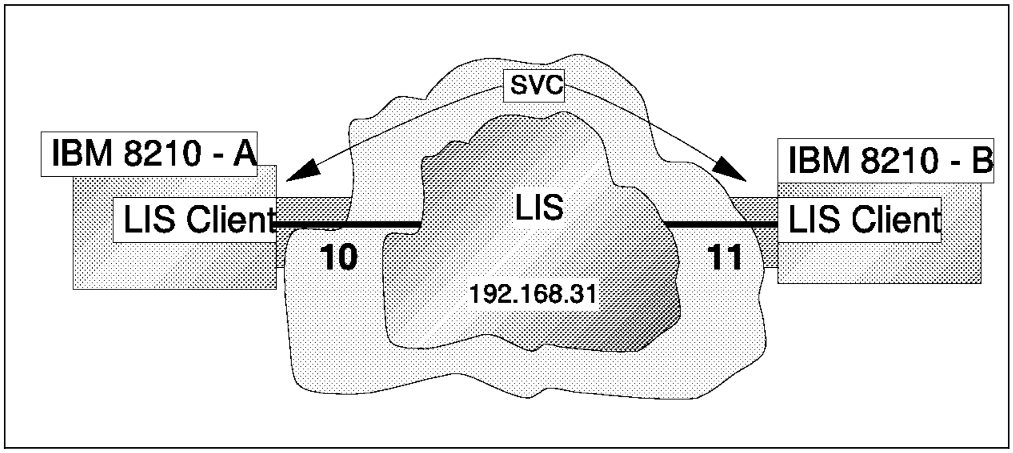


Figure 257. LIS to LIS Client Using SVC

Figure 257 depicts a configuration where an IBM 8210 Nways MSS Server contains a single LIS client that connects to another IBM 8210 Nways MSS Server. An SVC is being used to connect the two LIS Clients. This scenario is typically used when the customer receives the IBM 8210 Nways MSS Server and configures it for IP using SVC, but no ARP server is configured yet. Predefining an SVC connection means that there is no need for an ARP Server.

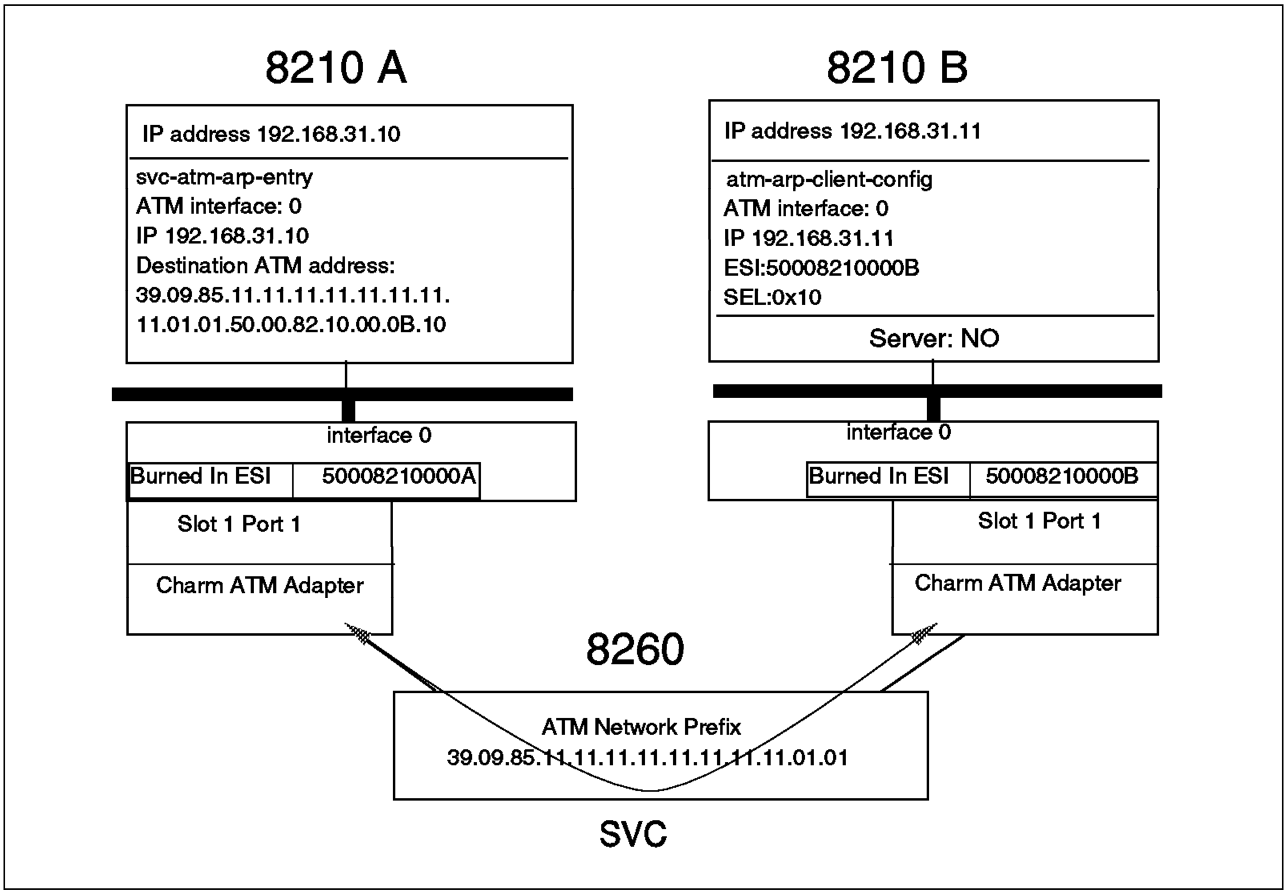


Figure 258. LIS Clients Using Static SVC - Parameter Overview

To realize the scenario depicted in Figure 257 we used the parameters depicted in Figure 258. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface on 8210 A

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 50.00.82.10.00.0A
ATM Interface Config>exit
ATM Config>exit
Config>
```

Figure 259. Set Up the ATM Interface

2 Configure the ATM interface on 8210 B

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 50.00.82.10.00.0B
ATM Interface Config>exit
ATM Config>exit
Config>
```

Figure 260. Set Up the ATM Interface

3 Define LIS client on 8210 A

Define IP address

```
Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.31.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 261 (Part 1 of 3). Configuring 8210 A

Configure the LIS client characteristics 1

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.32.11
This client is also a server? [No]: no
Refresh timeout (in minutes) [5]?
Enable auto-refresh? [No]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 50008210000A
Select ESI [1]? 2
Use internally assigned selector? [Yes]:
Selector Only, Range 00..FF [00]?
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Note:

1 LIS client definition is optional. Implicit LIS client, with default settings, will result if no atm-arp-client-configuration is entered.

Figure 261 (Part 2 of 3). Configuring 8210 A

Configure the SVC ARP entry

```
Config>protocol arp
ARP user configuration
ARP config>add svc-atm-arp-entry
Interface Number [0]? 0
Protocol [IP]?
Local client IP Address [0.0.0.0]? 192.168.31.10
Specify destination protocol address? [Yes]:no     1
Destination ATM Address []? :
39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.0B.00.10
ARP config>exit
Config>
```

Note:

1 Partner IP address will be learned dynamically

Figure 261 (Part 3 of 3). Configuring 8210 A

4 Configure LIS client on 8210 B

Define IP address

```
Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.31.11
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 262 (Part 1 of 2). Configuring 8210 B

Configure the LIS client on 8210 B

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.31.11
This client is also a server? [No]: no
Refresh timeout (in minutes) [5]?
Enable auto-refresh? [No]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 50008210000B
Select ESI [1]? 2
Use internally assigned selector? [Yes]: no   1
Selector Only, Range 00..FF [00]? 10
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Note:

1 User-defined selector simplifies definition of ATM address on 8210-A

Figure 262 (Part 2 of 2). Configuring 8210 B

5 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 263 (Part 1 of 2). Restart and Verify

Verify the IP configuration

```
*talk 5

CGW Operator Console

+protocol ip
IP>ping 192.168.31.11                                  1
PING 192.168.31.10 -> 192.168.31.11: 56 data bytes, ttl=64, every 1 sec.
56 data bytes from 192.168.31.11: icmp_seq=0. ttl=64. time=0. ms
56 data bytes from 192.168.31.11: icmp_seq=1. ttl=64. time=0. ms
56 data bytes from 192.168.31.11: icmp_seq=2. ttl=64. time=0. ms

----192.168.31.11 PING Statistics----
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0/0/0 ms
IP>exit
```

Note:

1 Pinging to remote LIS client (192.168.31.11)

Figure 263 (Part 2 of 2). Restart and Verify

11.4.12 IBM 8210 LIS to LIS Routing

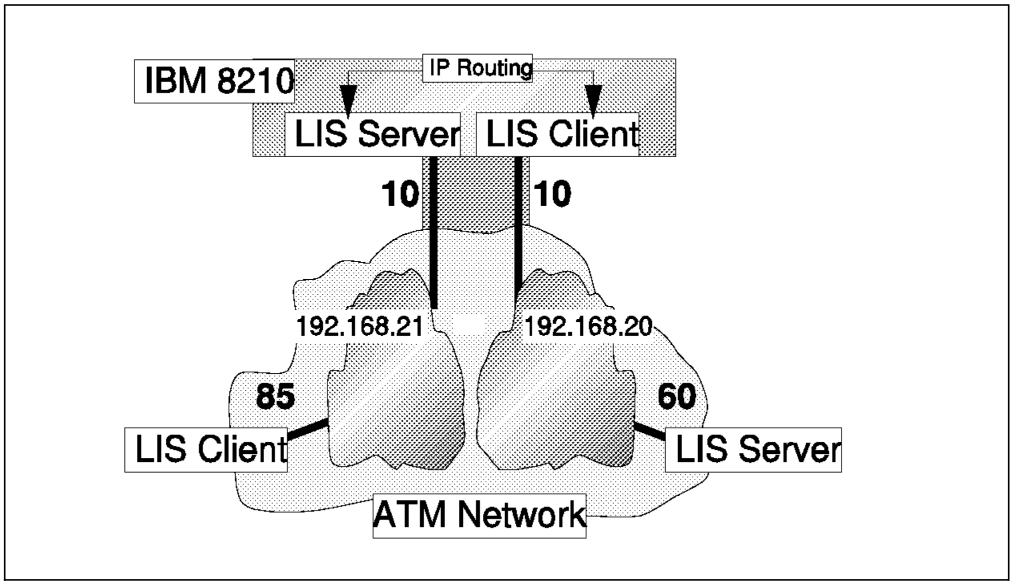


Figure 264. IP Routing between LISs

Figure 264 depicts a configuration where the IBM 8210 Nways MSS Server is connected to two LISs and performs IP routing between them. For the first (left-hand) LIS, the IBM 8210 Nways MSS Server performs both client and (ARP) server functions. For the other the IBM 8210 Nways MSS Server performs only client functions.

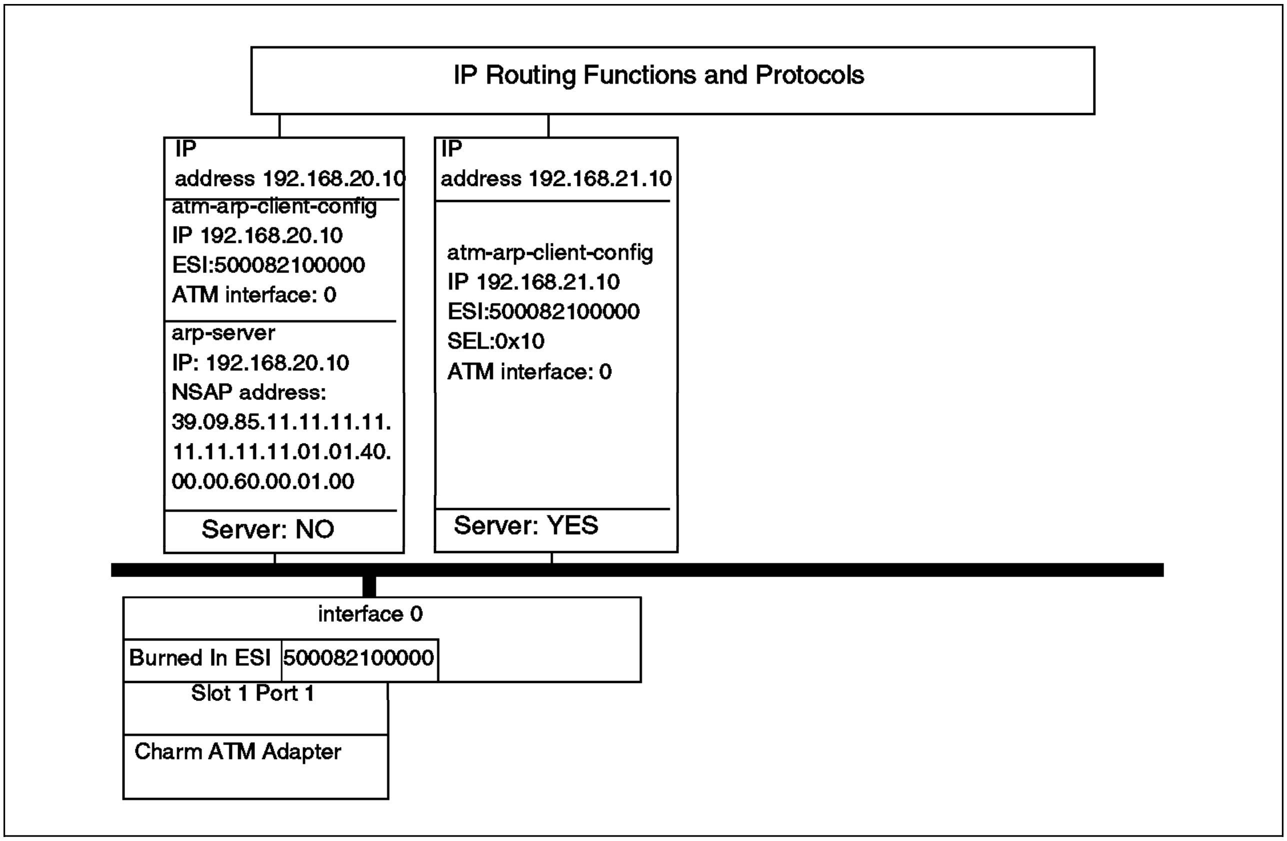


Figure 265. IP Routing between LISs - Parameter Overview

To achieve the scenario depicted in Figure 264 we used the parameters depicted in Figure 265. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 50.00.82.10.00.00
ATM Interface Config>exit
ATM Config>exit
Config>
```

Figure 266. Define ATM Interface

2 Configure LIS client 192.168.20.10

Configure IP on ATM interface

```
Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.20.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 267 (Part 1 of 3). 8210 LIS Client

Configure the LIS client

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.20.10
This client is also a server? [No]: no
Refresh timeout (in minutes) [5]?
Enable auto-refresh? [No]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 500082100000
Select ESI [1]? 2
Use internally assigned selector? [Yes]: yes
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>
```

Figure 267 (Part 2 of 3). 8210 LIS Client

Assign the ARP Server for the LIS client

```
ARP config>add arp-server private-nsapa
Local Client IP Address [0.0.0.0]? 192.168.20.10
Private NSAP Address: Specify 40 digits
ATM Address []? 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.60.00.01.00
ARP config>exit
Config>
```

Figure 267 (Part 3 of 3). 8210 LIS Client

3 Configure the ARP Server/LIS client 192.168.21.10

Configure IP on the ATM interface

```
Config>protocol ip
Internet protocol user configuration
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.21.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 268 (Part 1 of 2). ARP Server

Configure the ARP Server

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.21.10
This client is also a server? [No]: yes
Refresh timeout (in minutes) [20]?
Enable auto-refresh? [Yes]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 500082100000
Select ESI [1]? 2
Use internally assigned selector? [Yes]: no
Selector Only, Range 00..FF [00]? 10
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Figure 268 (Part 2 of 2). ARP Server

4 Restart 8210 and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 269 (Part 1 of 2). Restart and Verify

Display Interface Status

```
*talk 5

CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list all
                   *******  USERS *******

UserHandle  FrameSap    ATM Address
----------  --------  -----------------------------------------------------------
320B3E94    320B3EA0  39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.10
320B3CFC    320B3D08  39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.C8


                   *******  ADDRESSES *******

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.10
39.09.85.11.11.11.11.11.11.11.11.01.01.50.00.82.10.00.00.C8


                   *******  VCCS *******

 Conn    Conn                             Frames       Frames     Bytes        Bytes
 Handle  Type  VPI  VCI    FrameSap   Transmitted  Received   Transmitted  R′cvd
 ------  ----  ---  ----  ----------  -----------  ---------  -----------  ------
      9   P-P    0   404    320B3EA0            3        365          156    30503
      8   P-P    0   403    320B3EA0          341          1        28794      108
      7   P-P    0   402    320B3D08            4         41          276     3417
      1  SAAL    0     5           0           27         27          428      528
      2  ILMI    0    16           0           58         58         3033     3567

ATM Interface+
```

Figure 269 (Part 2 of 2). Restart and Verify

11.4.13 IBM 8210 LIS to Token-Ring ELAN IP Routing

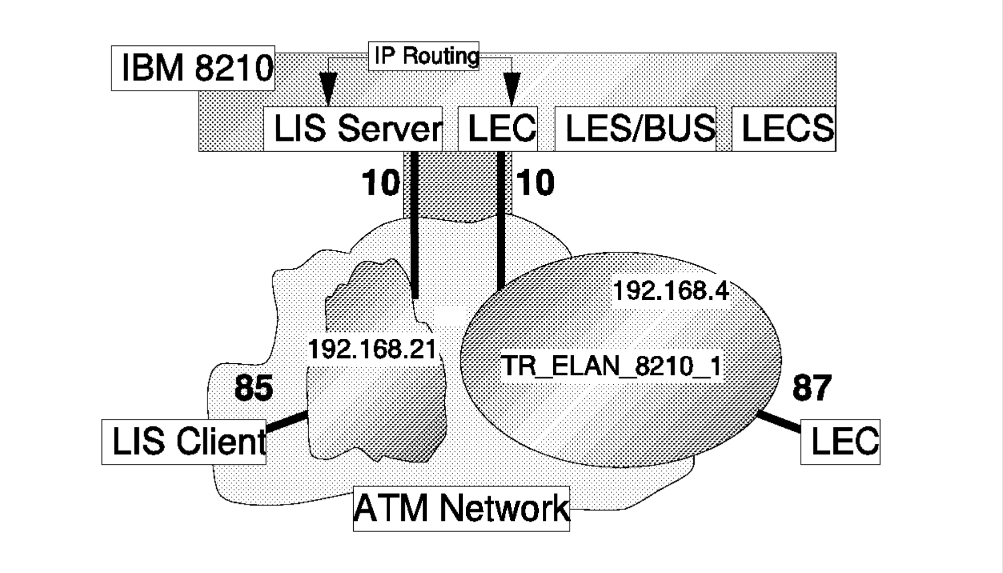


Figure 270. IP Routing between LIS and ELAN

Figure 270 depicts a configuration where the IBM 8210 Nways MSS Server is connected to a LIS and a token-ring ELANs, and is enabled for IP routing between them. The ELAN is controlled by a local LES/BUS. No security is enforced, and LE clients can join without connecting to the LECS first. An LECS has been defined to enable LE clients to join the ELAN. The LECS assigns LE clients to the ELAN using a name policy.

Although not critical for the IP routing function, we have activated BCM and SRM functions for IP on the token-ring ELAN.

Note: Both IP addresses provide access to the MSS Server′s management functions.

To realize the scenario depicted in Figure 270 we used the parameters depicted in Figure 271 on page 357. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 50.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 272. Define ATM Interface

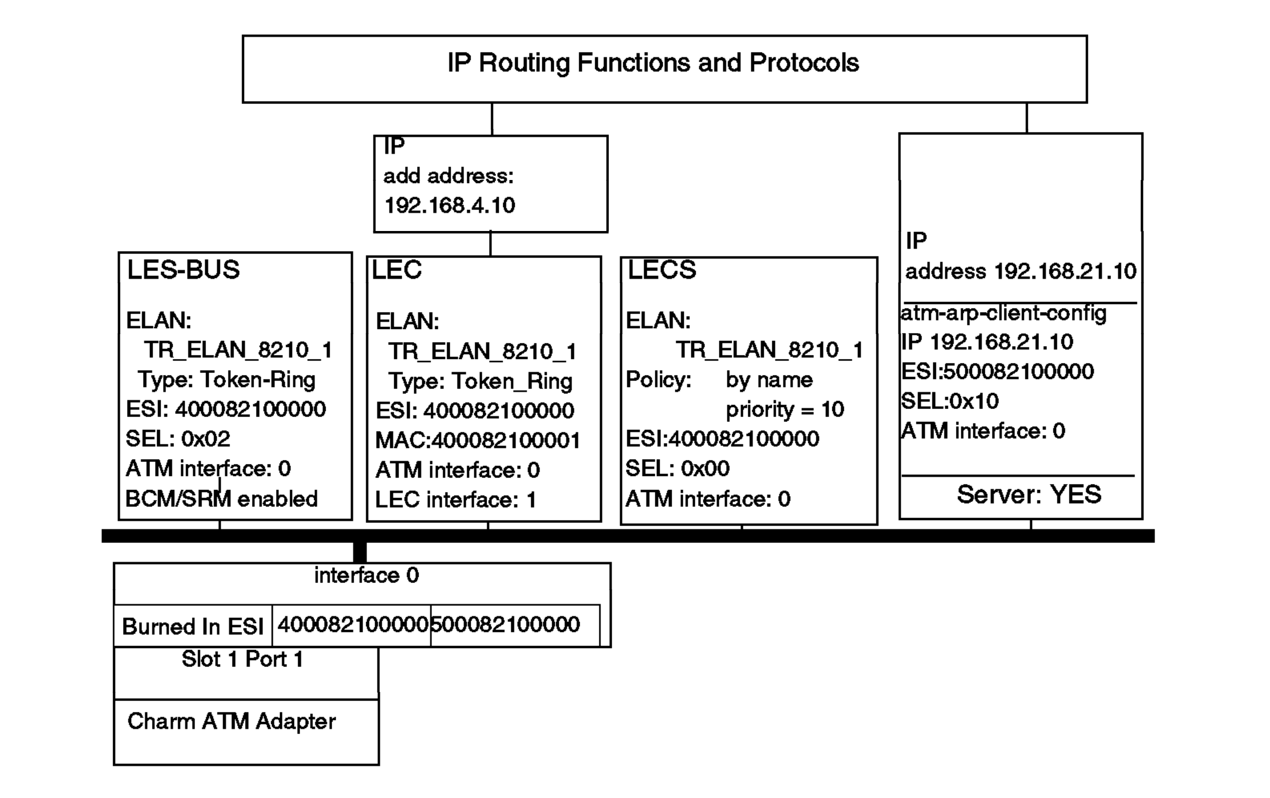


Figure 271. LIS and Token-Ring ELAN - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 Configure LECS

Add the LECS to the local administered ESI

```
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  ( 1) Use burned in ESI
  ( 2) 40.00.82.10.00.00
  ( 3) 50.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 273 (Part 1 of 4). LECS Definition

Add a token-ring ELAN definition to the LECS

```
LECS config>elan
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ’TR_ELAN_8210_1’ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 273 (Part 2 of 4). LECS Definition

Configure the ELAN definition, and set policy value

```
LECS ELANs config>select
  (  1) TR_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN ’TR_ELAN_8201_1’ selected for detailed configuration
Selected ELAN ’TR_ELAN_8210_1’>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_1
    added to ELAN ’TR_ELAN_8210_1’
Selected ELAN ’TR_ELAN_8210_1’>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1
ELAN name ’TR_ELAN_8210_1’
     bound to LES Local LES for: TR_ELAN_8210_1
Selection ″ELAN name add″ Complete
Selected ELAN ’TR_ELAN_8210_1’>exit
LECS ELANs config>exit
LECS config>
```

Figure 273 (Part 3 of 4). LECS Definition

Define general policies priority

```
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy 'byElanNm ' at priority 10
Selection "Add assignment policy" Complete
LECS POLICIES config>exit
LECS Config>exit
LE Services config>
```

Figure 273 (Part 4 of 4). LECS Definition

3 Configure the LES/BUS pair

```
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN 'TR_ELAN_8210_1'>add
Turn on Standard Event Logging for LES [yes]
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00
        (3) 50.00.82.10.00.00

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'TR_ELAN_8210_1'>enable bcm IP   1
LES-BUS config for ELAN 'TR_ELAN_8210_1'>enable source    2
LES-BUS config for ELAN 'TR_ELAN_8210_1'>exit
LE Services config>exit
ATM Config>
```

Note:

1 Enabling Broadcast Management for IP

2 Enabling Source Routing Management for IP

Figure 274. Define Token-Ring LES/BUS Pair

4 Configure the token-ring LEC

Add a token-ring LEC as a (logical) interface

```
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add token-ring
Added Emulated LAN as interface 1
LE Client config>
```

Figure 275 (Part 1 of 2). Token-Ring LEC Definition

Configure the (logical) interface

```
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_1
Token Ring Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Token Ring Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00
   (3) 50.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x3
Token Ring Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 275 (Part 2 of 2). Token-Ring LEC Definition

5 Configure IP on the token-ring LEC

```
Config>protocol ip
Internet protocol user configuration
IP config>add address 1
New address [0.0.0.0]? 192.168.4.10
Address mask [255.255.255.0]?
IP config>
```

Figure 276. Adding IP

6 Configure the ARP Server/LIS client

Configure IP on the ATM interface

```
IP config>add address
Which net is this address for [0]? 0
New address [0.0.0.0]? 192.168.21.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Figure 277 (Part 1 of 2). ARP Server

Configure the ARP server

```
Config>protocol arp
ARP user configuration
ARP config>add atm-arp-client-configuration
Interface Number [0]? 0
Protocol [IP]?
Client IP Address [0.0.0.0]? 192.168.21.10
This client is also a server? [No]: yes
Refresh timeout (in minutes) [20]?
Enable auto-refresh? [Yes]:
Refresh by InAtmArp? [Yes]:
  (  1) Use burned in ESI
  (  2) 400082100000
  (  3) 500082100000
Select ESI [1]? 3
Use internally assigned selector? [Yes]: no
Selector Only, Range 00..FF [00]? 10
Validate PCR for best effort VCCs? [No]:
Maximum Reserved Bandwidth for incoming VCCs (Kbps) [0]?
Use Best Effort Service for Control VCCs? [Yes]:
Peak Cell Rate of outbound control VCCs (Kbps) [0]?
Sustained Cell Rate of outbound control VCCs (Kbps) [0]?
Use Best Effort Service for Data VCCs? [Yes]:
Peak Cell Rate of outbound Data VCCs (Kbps) [0]?
Sustained Cell Rate of outbound Data VCCs (Kbps) [0]?
Max SDU size (bytes) [9188]?
ARP config>exit
Config>
```

Figure 277 (Part 2 of 2). ARP Server

7 Restart 8210

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 278. Restart MSS Server

Note: Verify the configuration by ping′ing between LIS and ELAN attached stations.

11.4.14 IBM 8210 Ethernet to Token-Ring ELAN IP Routing

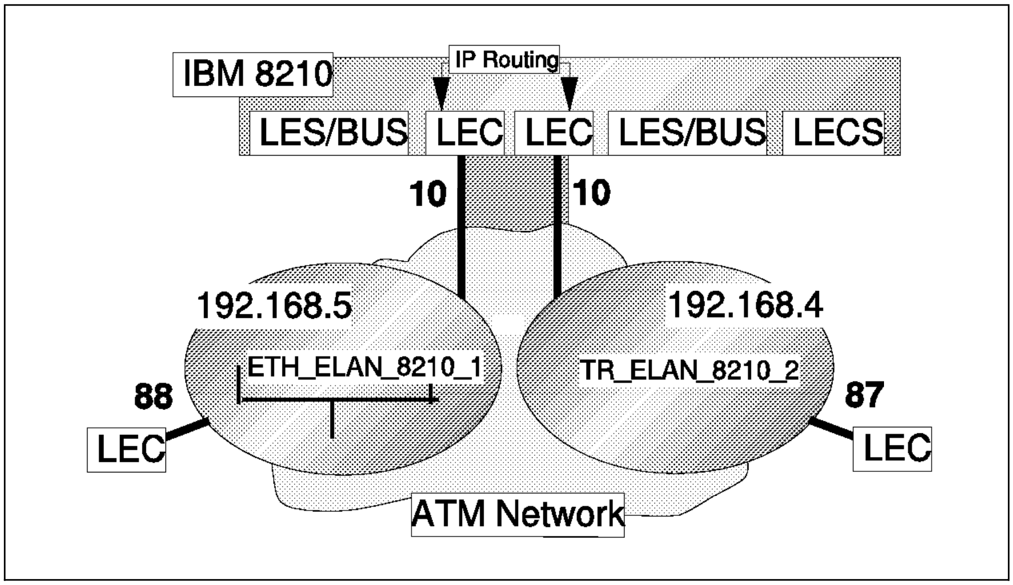


Figure 279. IP Routing between ELANs

Figure 279 depicts a configuration where the IBM 8210 Nways MSS Server is connected to two ELANs, token-ring and Ethernet respectively, and is enabled for IP routing between them. Both ELANs are controlled by local LES/BUSs. An LECS has been defined to enable LE clients to join the ELANs. No security is enforced, and LE clients can join without connecting to the LECS first. The LECS assigns LE clients to the ELAN using a name policy.

Although not critical for the IP routing function, we have enabled BCM functions for IP on both ELANs. In addition, we have enabled SRM on the token-ring ELAN.

Note: The management functions can be accessed via either of the IP addresses associated with the LE clients.

To realize the scenario depicted in Figure 279 we used the parameters depicted in Figure 280 on page 363. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 281. Configure ATM Interface

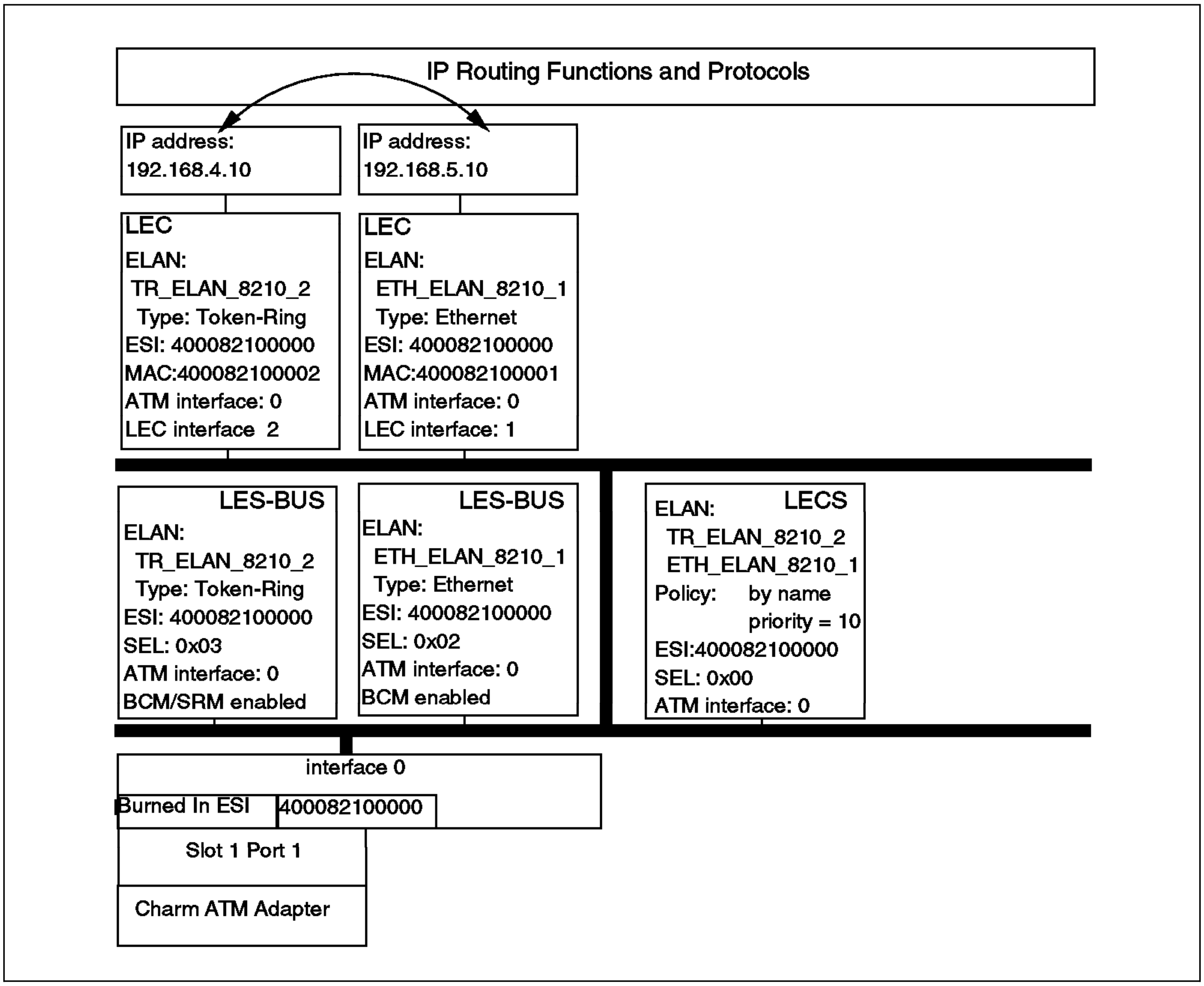


Figure 280. IP Routing between ELANs - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 **Configure the LECS**

Add the LECS

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 282 (Part 1 of 6). Configure LECS

Add ELAN ′ETH_ELAN_8210_1′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? ETH_ELAN_8210_1
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 1
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 1
ELAN ′ETH_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 282 (Part 2 of 6). Configure LECS

Add ELAN ′TR_ELAN_8210_2′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_2
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_2′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 282 (Part 3 of 6). Configure LECS

Configure the ′ETH_ELAN_8210_1′ ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_1
  (  2) TR_ELAN_8210_2
Choice of ELAN [1]? 1
ELAN ′ETH_ELAN_8210_1′ selected for detailed configuration
Selected ELAN ′ETH_ELAN_8210_1′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: ETH_ELAN_8210_1
    added to ELAN ′ETH_ELAN_8210_1′
Selected ELAN ′ETH_ELAN_8210_1′>
Selected ELAN ′ETH_ELAN_8210_1′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: ETH_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? ETH_ELAN_8210_1
ELAN name ′ETH_ELAN_8210_1′
     bound to LES Local LES for: ETH_ELAN_8210_1
Selection ″ELAN name add″ Complete
Selected ELAN ′ETH_ELAN_8210_1′>exit
LECS ELANs config>
```

Figure 282 (Part 4 of 6). Configure LECS

Configure the 'TR_ELAN_8210_2' ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_1
  (  2) TR_ELAN_8210_2
Choice of ELAN [1]? 2
ELAN 'TR_ELAN_8210_2' selected for detailed configuration
Selected ELAN 'TR_ELAN_8210_2'>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_2
    added to ELAN 'TR_ELAN_8210_2'
Selected ELAN 'TR_ELAN_8210_2'>
Selected ELAN 'TR_ELAN_8210_2'>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_2



Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_2                      .
ELAN name 'TR_ELAN_8210_2'
     bound to LES Local LES for: TR_ELAN_8210_2
Selection "ELAN name add" Complete
Selected ELAN 'TR_ELAN_8210_2'>exit
LECS ELANs config>exit
LECS config>
```

Figure 282 (Part 5 of 6). Configure LECS

Enable LECS policies

```
LECS config>
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete

LECS POLICIES config>list all

Policy Listing...

Enabled  Priority  Type
=======  ========  ==========
    Yes        10  byElanNm
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 282 (Part 6 of 6). Configure LECS

3 Configure the LES/BUS

Add the LES/BUS for the 'ETH_ELAN_8210_1' ELAN

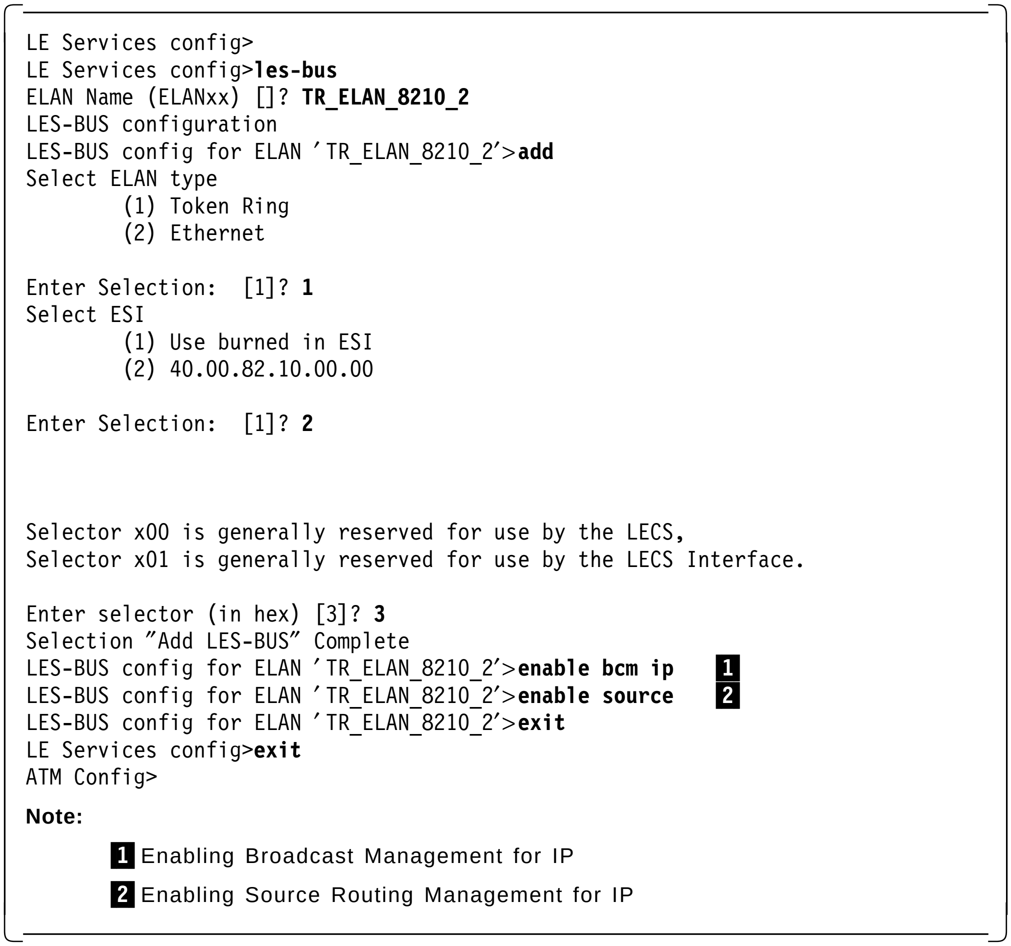
```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_2
LES-BUS configuration
LES-BUS config for ELAN 'ETH_ELAN_8210_1'>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 2
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2


Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'ETH_ELAN_8210_1'>enable bcm ip    1
LES-BUS config for ELAN 'ETH_ELAN_8210_1'>exit
LE Services config>
```

Note:

1 Enabling Broadcast Management for IP

Figure 283 (Part 1 of 2). Configure LES/BUS

Add the LES/BUS for the 'TR_ELAN_8210_2' ELAN



```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_2
LES-BUS configuration
LES-BUS config for ELAN 'TR_ELAN_8210_2'>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2



Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [3]? 3
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'TR_ELAN_8210_2'>enable bcm ip    1
LES-BUS config for ELAN 'TR_ELAN_8210_2'>enable source    2
LES-BUS config for ELAN 'TR_ELAN_8210_2'>exit
LE Services config>exit
ATM Config>
```

Note:

1 Enabling Broadcast Management for IP

2 Enabling Source Routing Management for IP

Figure 283 (Part 2 of 2). Configure LES/BUS

4 Configure the LEC

Configure an LEC for the ′ETH_ELAN_8210_1′ ELAN

```
ATM Config>
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8210_1
Ethernet Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x4
Ethernet Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Ethernet Forum Compliant LEC Config>
Ethernet Forum Compliant LEC Config>exit
LE Client config>
```

Figure 284 (Part 1 of 2). Configure LEC

Configure an LEC for the ′TR_ELAN_8210_2′ ELAN

```
LE Client config>
LE Client config>add token
Added Emulated LAN as interface 2
LE Client config>config
Emulated LAN interface number [1]? 2
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_2
Token Ring Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x5
Token Ring Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.02
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 284 (Part 2 of 2). Configure LEC

5 **Configure IP**

Configure IP on both interfaces

```
Config>protocol ip
Internet protocol user configuration
IP config>add address 1                          1
New address [0.0.0.0]? 192.168.5.10      2
Address mask [255.255.255.0]? 255.255.255.0
IP config>add address 2
New address [0.0.0.0]? 192.168.4.10      3
Address mask [255.255.255.0]? 255.255.255.0
IP config>
```

Note:

1 Adding an IP address to an interface automatically enables IP

2 This is the IP address of the internal Ethernet LEC (interface 1)

3 This is the IP address of the internal token-ring LEC (interface 2)

Figure 285 (Part 1 of 2). Configure IP

List the IP configuration

```
IP config>list all
Interface addresses
IP addresses for each interface:
   intf  0                                     IP disabled on this interface
   intf  1   192.168.5.10     255.255.255.0    Local wire broadcast, fill 1
   intf  2   192.168.4.10     255.255.255.0    Local wire broadcast, fill 1

Routing



Protocols
BOOTP forwarding: disabled
IP Time-to-live: 64
Source Routing: enabled
Echo Reply: enabled
Directed broadcasts: enabled
ARP subnet routing: disabled
ARP network routing: disabled
Per-packet-multipath: disabled
OSPF: disabled
BGP: disabled
RIP: disabled

IP config>exit
Config>
```

Figure 285 (Part 2 of 2). Configure IP

6 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 286 (Part 1 of 2). Restart and Verify

Verify the IP configuration

```
*talk 5

CGW Operator Console

+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list address

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.02
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.03
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.00
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.04   1
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.05   1
ATM Interface+exit
ATM+exit
*
```

Note:

1 The LECs ATM address are active

Figure 286 (Part 2 of 2). Restart and Verify

Note: Verify the configuration by ping′ing between token-ring and Ethernet ELAN attached stations.

11.4.15 IBM 8210 Token-Ring to Ethernet ELAN IPX Routing

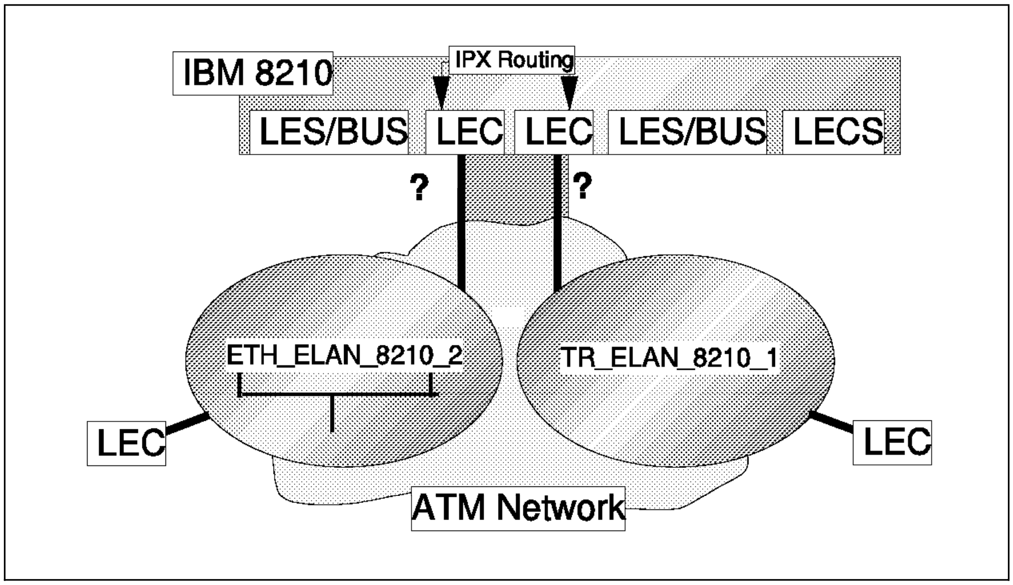


Figure 287. IPX Routing between ELANs

Figure 287 depicts a configuration where the IBM 8210 Nways MSS Server is connected to two ELANs, token-ring and Ethernet respectively, and is enabled for IPX routing between them. Both ELANs are controlled by local LES/BUSs. An LECS has been defined to enable LE clients to join the ELANs. No security is enforced, and LE clients can join without connecting to the LECS first. The LECS assigns LE clients to ELANs based on ELAN name. To enable IP access to the management functions, we have configured the IP host services.

Although not critical for the IPX routing function, we have enabled BCM functions for IPX on both ELANs.

Note: If access to the management functions is required, configure the IP host services as well.

To realize the scenario depicted in Figure 287 we used the parameters depicted in Figure 288 on page 374. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 289. Configure ATM Interface

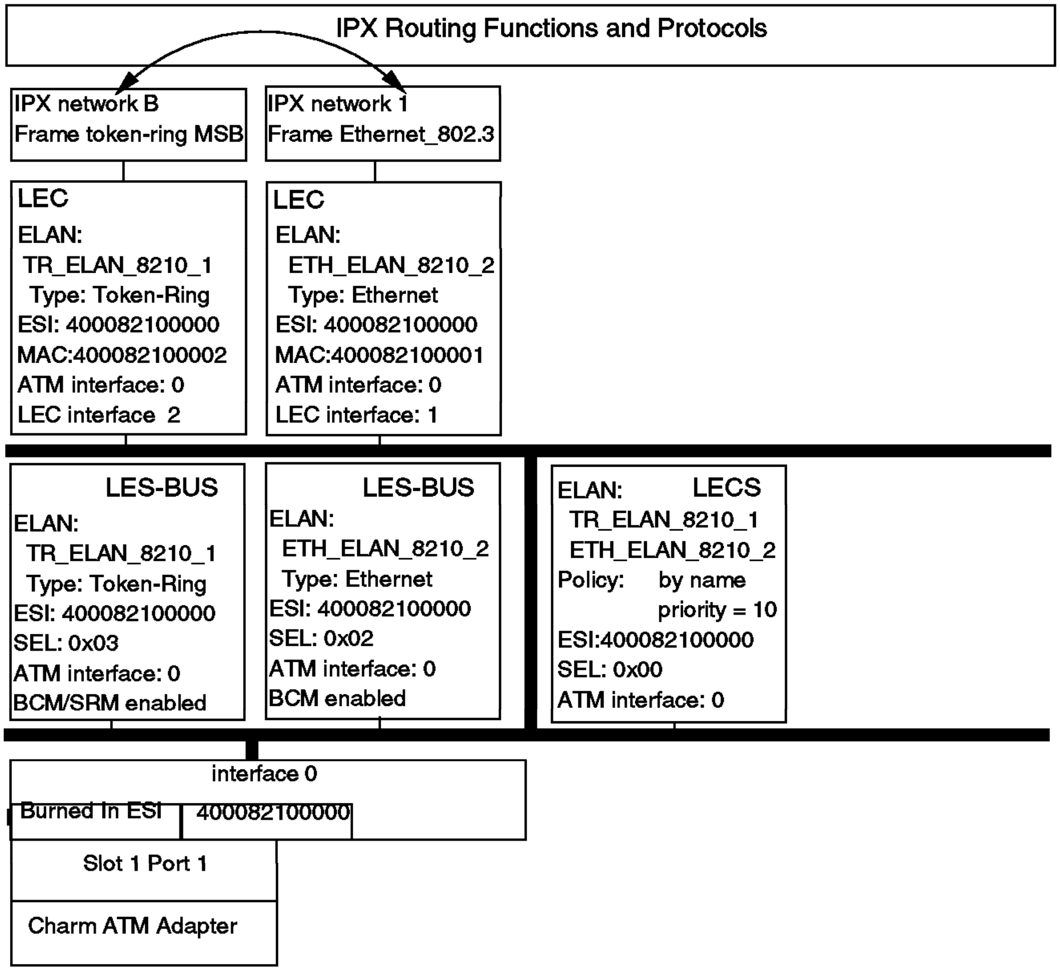


Figure 288. IPX Routing between ELANs - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 **Configure the LECS**

Add the LECS

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 290 (Part 1 of 6). Configure LECS

Add ELAN ′ETH_ELAN_8210_2′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? ETH_ELAN_8210_2
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 1
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 1
ELAN ′ETH_ELAN_8210_2′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 290 (Part 2 of 6). Configure LECS

Add ELAN ′TR_ELAN_8210_1′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 290 (Part 3 of 6). Configure LECS

Configure the ′ETH_ELAN_8210_2′ ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_2
  (  2) TR_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN ′ETH_ELAN_8210_2′ selected for detailed configuration
Selected ELAN ′ETH_ELAN_8210_2′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: ETH_ELAN_8210_2
    added to ELAN ′ETH_ELAN_8210_2′
Selected ELAN ′ETH_ELAN_8210_2′>
Selected ELAN ′ETH_ELAN_8210_2′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: ETH_ELAN_8210_2

Enter Selection:  [1]? 1
ELAN name []? ETH_ELAN_8210_2
ELAN name ′ETH_ELAN_8210_2′
     bound to LES Local LES for: ETH_ELAN_8210_2
Selection ″ELAN name add″ Complete
Selected ELAN ′ETH_ELAN_8210_2′>exit
LECS ELANs config>
```

Figure 290 (Part 4 of 6). Configure LECS

Configure the 'TR_ELAN_8210_1' ELAN, and configure policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_2
  (  2) TR_ELAN_8210_1
Choice of ELAN [1]? 2
ELAN 'TR_ELAN_8210_1' selected for detailed configuration
Selected ELAN 'TR_ELAN_8210_1'>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_1
    added to ELAN 'TR_ELAN_8210_1'
Selected ELAN 'TR_ELAN_8210_1'>
Selected ELAN 'TR_ELAN_8210_1'>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_1


Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1                      .
ELAN name 'TR_ELAN_8210_1'
     bound to LES Local LES for: TR_ELAN_8210_1
Selection "ELAN name add" Complete
Selected ELAN 'TR_ELAN_8210_1'>exit
LECS ELANs config>exit
LECS config>
```

Figure 290 (Part 5 of 6). Configure LECS

Enable LECS policies

```
LECS config>
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete

LECS POLICIES config>list all

Policy Listing...

Enabled  Priority  Type
=======  ========  ==========
    Yes        10  byElanNm
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 290 (Part 6 of 6). Configure LECS

3 Configure the LES/BUS

Add the LES/BUS for the 'ETH_ELAN_8210_2' ELAN

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_2
LES-BUS configuration
LES-BUS config for ELAN 'ETH_ELAN_8210_2'>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 2
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2


Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'ETH_ELAN_8210_2'>enable bcm ipx   1
LES-BUS config for ELAN 'ETH_ELAN_8210_2'>exit
LE Services config>
```

Note:

1 Enabling Broadcast Management for IPX

Figure 291 (Part 1 of 2). Configure LES/BUS

Add the LES/BUS for the ′TR_ELAN_8210_1′ ELAN

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN ′TR_ELAN_8210_1′>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2



Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [3]? 3
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′TR_ELAN_8210_1′>enable bcm ipx   1
LES-BUS config for ELAN ′TR_ELAN_8210_1′>enable source    2
LES-BUS config for ELAN ′TR_ELAN_8210_1′>exit
LE Services config>exit
ATM Config>
```

Note:

1 Enabling Broadcast Management for IPX

2 Enabling Source Routing Management for IPX

Figure 291 (Part 2 of 2). Configure LES/BUS

4 Configure the LEC

Configure an LEC for the ′ETH_ELAN_8210_2′ ELAN

```
ATM Config>
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8210_2
Ethernet Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x4
Ethernet Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Ethernet Forum Compliant LEC Config>
Ethernet Forum Compliant LEC Config>exit
LE Client config>
```

Figure 292 (Part 1 of 2). Configure LEC

Configure an LEC for the ′TR_ELAN_8210_1′ ELAN

```
LE Client config>
LE Client config>add token
Added Emulated LAN as interface 2
LE Client config>config
Emulated LAN interface number [1]? 2
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_1
Token Ring Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x5
Token Ring Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.02
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 292 (Part 2 of 2). Configure LEC

5 Configure IPX

Configure IPX on both interfaces

```
Config>
Config>protocol ipx
IPX protocol user configuration
IPX config>enable ipx                    1
IPX config>enable interface              2
Which interface [0]? 1
Configure an IPX network number for this interface.
Network number in hex [1]? 1   3
IPX config>enable interface
Which interface [0]? 2
Configure an IPX network number for this interface.
Network number in hex [1]? B   4
```

Note:

1 IPX is being enabled globally

2 This enables IPX per interface

3 This is the network number of the Ethernet internal LEC (interface 1)

4 This is the network number of the token-ring internal LEC (interface 2)

Figure 293 (Part 1 of 2). Configure IPX

List IPX configuration

```
IPX config>
IPX config>list all

IPX globally                enabled
Host number (serial line)   000000000000
Router Name (IPXWAN)
NodeID (IPXWAN)             0
Maximum networks                  32
Maximum total route entries       32
Maximum routes per dest. network  1
Maximum services                  32
Maximum Network Cache entries     64
Maximum Local Cache entries       64

List of configured interfaces:
                Frame                   SAP nearest  Split
Ifc  IPX net #  Encapsulation           server reply Horizon      IPXWAN

  1          1  ETHERNET_802.3          Enabled      Enabled      N/A
  2          B  TOKEN-RING        MSB   Enabled      Enabled      N/A


RIP/SAP Timer Intervals
Ifc  IPX net #       SAP Interval(Minutes)   RIP Interval(Minutes)
  1          1                  1                      1
  2          B                  1                      1
IPX SAP Filter is: disabled
No IPX SAP Filter records in configuration.
IPX Access Controls are: disabled
No IPX Access Control records in configuration.
IPX config>exit
Config>
```

Figure 293 (Part 2 of 2). Configure IPX

6 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 294 (Part 1 of 2). Restarting and Verify

Verify the IPX configuration

```
+
+network 0
ATM Console
ATM+interface
ATM Interface Console
ATM Interface+list address

                         ATM Address
          Network Prefix                     ESI         SEL
-------------------------------------- ----------------- --
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.02
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.03
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.00
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.04    1
39.09.85.11.11.11.11.11.11.11.11.01.03.40.00.82.10.00.00.05    1
ATM Interface+exit
ATM+exit
+protocol ipx
IPX>slist          2
State Typ Service Name                 Hops   Age     Net  /   Host    / Sock
 SAP 009A NETWARE_SERVER                 1    0:35        B/400095950002/9100

1 entries used out of 32
IPX>exit
+
*
```

Note:

1 The LECs ATM address are active

2 Slist′ing to verify if IPX is being routed

Figure 294 (Part 2 of 2). Restarting and Verify

11.4.16 IBM 8210 Source-Route Bridging (SRB)

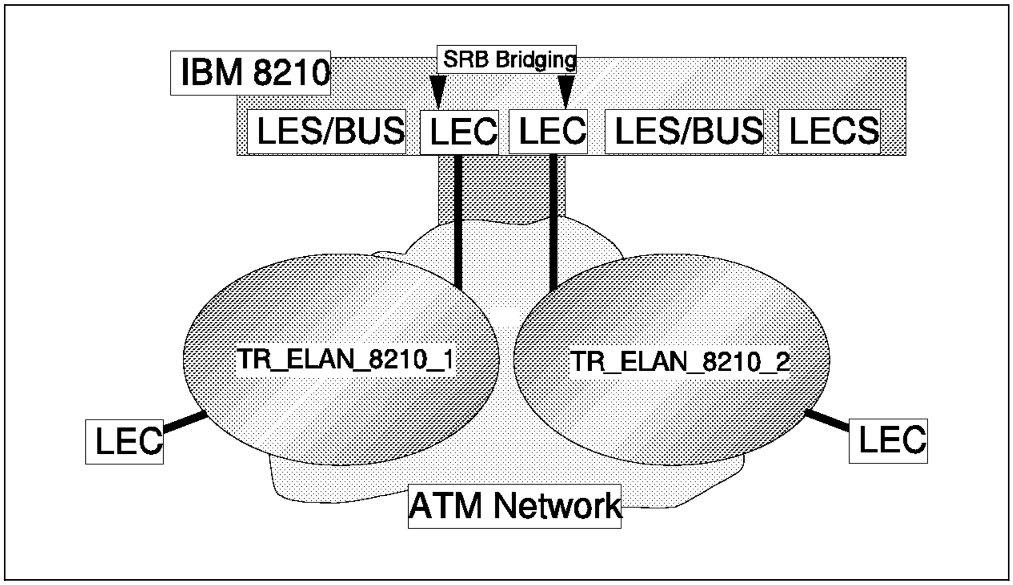


Figure 295. Source-Route Bridging

Figure 295 depicts a configuration where the IBM 8210 Nways MSS Server is connected to two token-ring ELANs and is enabled for SRB bridging between them.

Both ELANs are controlled by local LES/BUSs. An LECS has been defined to enable LE clients to join the ELANs. The LECS assigns LE clients to the ELANs using a name policy. No security is enforced, and LE clients can join without connecting to the LECS first.

To enable IP access to the MSS Server′s management functions we have configured the IP host services.

Although not critical for the bridging function, we have enabled BCM and SRM functions for IP, IPX, and NetBIOS on both token-ring ELANs.

To realize the scenario depicted in Figure 295 we used the parameters depicted in Figure 296 on page 385. For the parameters not shown we used default values. The configuration steps required are:

1 Configure the ATM interface

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 297. Configure ATM Interface

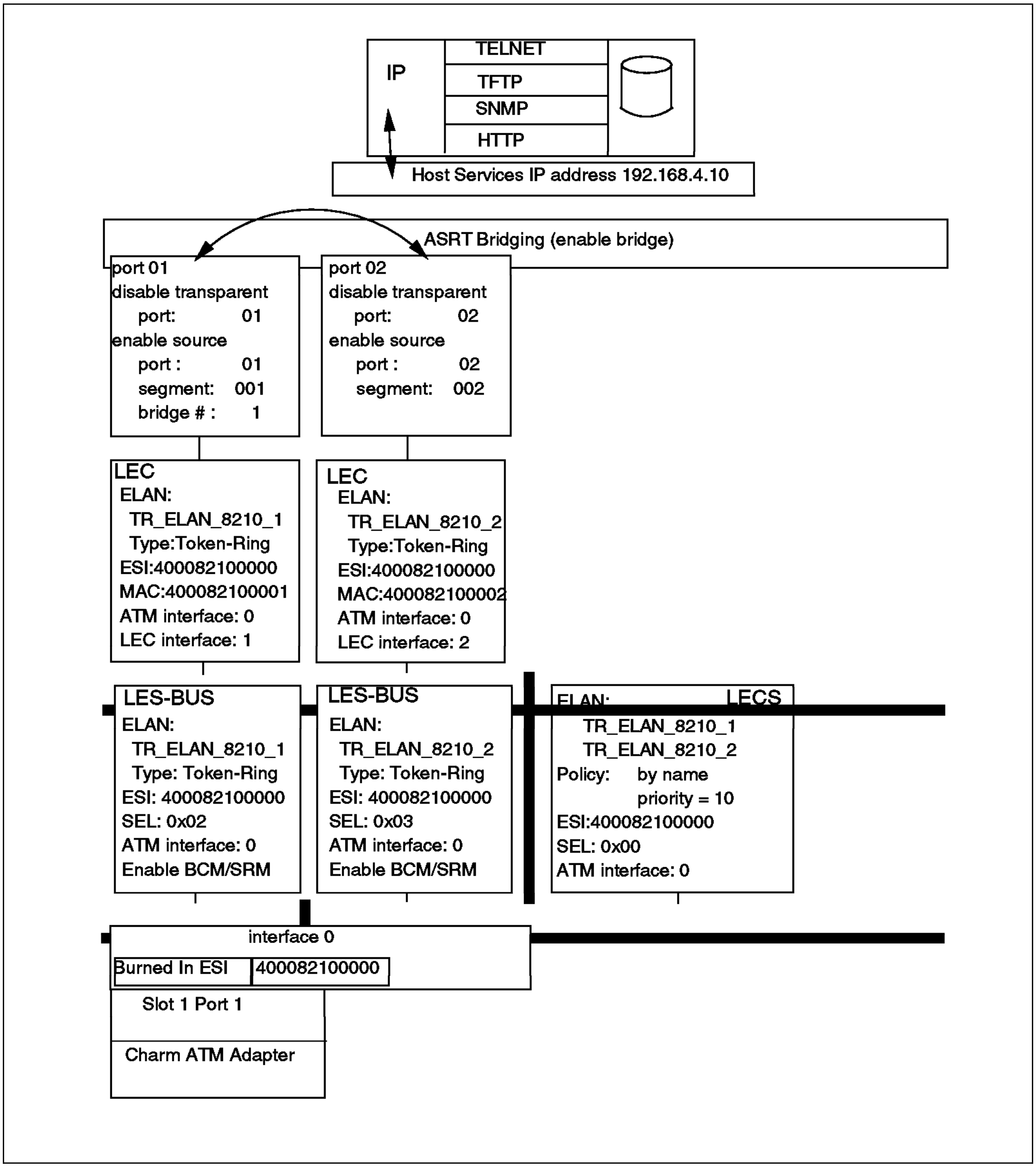


Figure 296. Source-Route Bridging - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 **Configure the LECS**

Add the LECS

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 298 (Part 1 of 6). Configure LECS

Add ELAN ′TR_ELAN_8210_1′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 298 (Part 2 of 6). Configure LECS

Add ELAN 'TR_ELAN_8210_2' to the LECS

```
LECS ELANs config>
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_2
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2

ELAN 'TR_ELAN_8210_2' added
Selection "ELAN addition" Complete
LECS ELANs config>
```

Figure 298 (Part 3 of 6). Configure LECS

Configure the 'TR_ELAN_8210_1' ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) TR_ELAN_8210_1
  (  2) TR_ELAN_8210_2
Choice of ELAN [1]? 1
ELAN 'TR_ELAN_8210_1' selected for detailed configuration
Selected ELAN 'TR_ELAN_8210_1'>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_1
    added to ELAN 'TR_ELAN_8210_1'
Selected ELAN 'TR_ELAN_8210_1'>
Selected ELAN 'TR_ELAN_8210_1'>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1
ELAN name 'TR_ELAN_8210_1'
     bound to LES Local LES for: TR_ELAN_8210_1
Selection "ELAN name add" Complete
Selected ELAN 'TR_ELAN_8210_1'>exit
LECS ELANs config>
```

Figure 298 (Part 4 of 6). Configure LECS

Configure the 'TR_ELAN_8210_2' ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) TR_ELAN_8210_1
  (  2) TR_ELAN_8210_2
Choice of ELAN [1]? 2
ELAN 'TR_ELAN_8210_2' selected for detailed configuration
Selected ELAN 'TR_ELAN_8210_2'>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_2
    added to ELAN 'TR_ELAN_8210_2'
Selected ELAN 'TR_ELAN_8210_2'>
Selected ELAN 'TR_ELAN_8210_2'>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_2


Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_2                      .
ELAN name 'TR_ELAN_8210_2'
     bound to LES Local LES for: TR_ELAN_8210_2
Selection "ELAN name add" Complete
Selected ELAN 'TR_ELAN_8210_2'>exit
LECS ELANs config>exit
LECS config>
```

Figure 298 (Part 5 of 6). Configure LECS

Enable LECS policies

```
LECS config>
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete

LECS POLICIES config>list all

Policy Listing...

Enabled  Priority  Type
=======  ========  ==========
    Yes        10  byElanNm
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 298 (Part 6 of 6). Configure LECS

3 Configure the LES/BUS

Add the LES/BUS to the ′TR_ELAN_8210_1′ ELAN

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN ′TR_ELAN_8210_1′>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2


Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′TR_ELAN_8210_1′>enable bcm all   1
LES-BUS config for ELAN ′TR_ELAN_8210_1′>enable source    2
LES-BUS config for ELAN ′TR_ELAN_8210_1′>exit
LE Services config>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

2 Enabling Source Routing Management for NetBIOS, IP and IPX

Figure 299 (Part 1 of 2). Configure LES/BUS

Add the LES/BUS to the ′TR_ELAN_8210_2′ ELAN

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_2
LES-BUS configuration
LES-BUS config for ELAN ′TR_ELAN_8210_2′>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2


Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [3]? 3
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′TR_ELAN_8210_2′>enable bcm all   1
LES-BUS config for ELAN ′TR_ELAN_8210_2′>enable source    2
LES-BUS config for ELAN ′TR_ELAN_8210_2′>exit
LE Services config>exit
ATM Config>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

2 Enabling Source Routing Management

Figure 299 (Part 2 of 2). Configure LES/BUS

4 Configure the LEC

Configure an LEC for ′TR_ELAN_8210_1′ ELAN

```
ATM Config>
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add token
Added Emulated LAN as interface 1
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_1
Token Ring Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x4
Token Ring Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Token Ring Forum Compliant LEC Config>
Token Ring Forum Compliant LEC Config>exit
LE Client config>
```

Figure 300 (Part 1 of 2). Configure LEC

Configure an LEC for ′TR_ELAN_8210_2′ ELAN

```
LE Client config>
LE Client config>add token
Added Emulated LAN as interface 2
LE Client config>config
Emulated LAN interface number [1]? 2
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_2
Token Ring Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x5
Token Ring Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.02
Token Ring Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 300 (Part 2 of 2). Configure LEC

5 Configure the source-route bridge

Enable and configure bridging

```
Config>
Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>enable bridge
ASRT config>list port
Port ID (dec)    : 128:01, (hex): 80-01              1
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only         2
Assoc Interface  : 1
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:02, (hex): 80-02              1
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only         2
Assoc Interface  : 2
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

ASRT config>disable transparent 1          3
ASRT config>disable transparent 2          3
ASRT config>enable source-routing 1        4
Segment Number for the port in hex(1 - FFF) [  1]? 001
Bridge number in hex (0 - 9, A - F) [0]? 1
ASRT config>enable source-routing 2        4
Segment Number for the port in hex(1 - FFF) [  1]? 002
```

Note:

1 These are the port numbers assigned to each interface and used for the bridge configuration.

2 Transparent bridging is by default enabled when you enable bridging

3 Disabling transparent bridging because we want a pure SRB

4 Enabling source routing bridge

Figure 301 (Part 1 of 2). Configure Source-Route Bridge

List bridge configuration

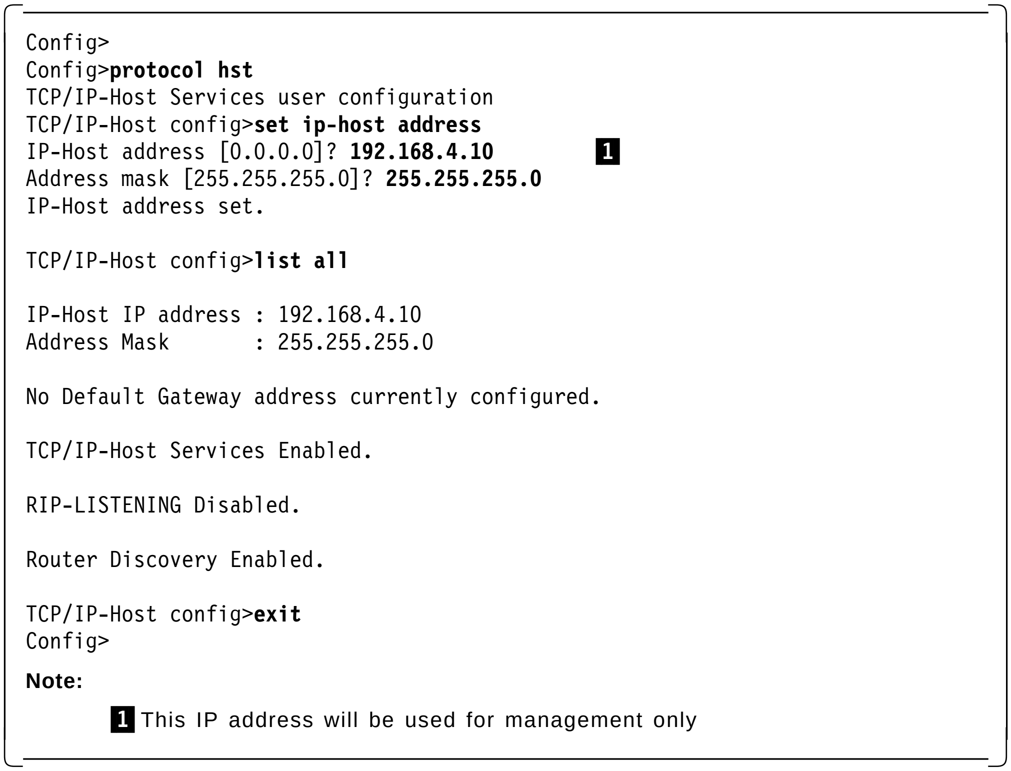
```
ASRT config>list bridge

                Source Routing Transparent Bridge Configuration
                ===============================================

Bridge:                  Enabled                 Bridge Behavior: SRB
                   +----------------------------+
-------------------| SOURCE ROUTING INFORMATION |-------------------------------

                   +----------------------------+
Bridge Number:           01                      Segments:          2
Max ARE Hop Cnt:         14                      Max STE Hop cnt:   14
1:N SRB:                 Not Active              Internal Segment:  0x000
LF-bit interpret:        Extended
                   +------------------+
-------------------| SR-TB INFORMATION |----------------------------------------

                   +------------------+
SR-TB Conversion:        Disabled
TB-Virtual Segment:      0x000                   MTU of TB-Domain:  0
                   +------------------------------------+
-------------------| SPANNING TREE PROTOCOL INFORMATION |-----------------------

                   +------------------------------------+
Bridge Address:          Default                 Bridge Priority:   32768/0x8000
STP Participation:       IBM-SRB proprietary
                   +-------------------------+
-------------------| TRANSLATION INFORMATION |----------------------------------

                   +-------------------------+
FA<=>GA Conversion:      Enabled                 UB-Encapsulation:  Disabled
                   +------------------+
-------------------| PORT INFORMATION |-----------------------------------------

                   +------------------+
Number of ports added: 2
Port:   1      Interface:       1      Behavior:    SRB Only   STP:  Enabled  1
Port:   2      Interface:       2      Behavior:    SRB Only   STP:  Enabled  1


ASRT config>exit
Config>
```

Note:

1 SRB is enabled on both interfaces

Figure 301 (Part 2 of 2). Configure Source-Route Bridge

6 Configure IP Host Services



```
Config>
Config>protocol hst
TCP/IP-Host Services user configuration
TCP/IP-Host config>set ip-host address
IP-Host address [0.0.0.0]? 192.168.4.10        1
Address mask [255.255.255.0]? 255.255.255.0
IP-Host address set.

TCP/IP-Host config>list all

IP-Host IP address : 192.168.4.10
Address Mask       : 255.255.255.0

No Default Gateway address currently configured.

TCP/IP-Host Services Enabled.

RIP-LISTENING Disabled.

Router Discovery Enabled.

TCP/IP-Host config>exit
Config>
```

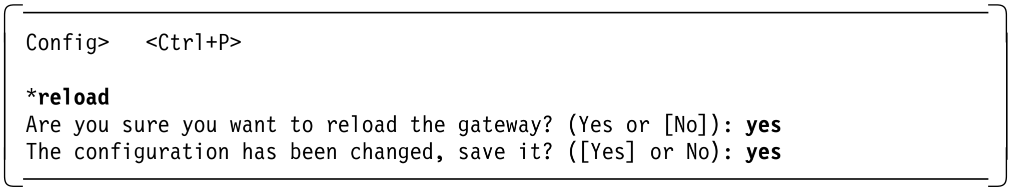
Note:

1 This IP address will be used for management only

Figure 302. Configure IP Host Services

7 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration



```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 303 (Part 1 of 2). Restart and Verify

```
Verify the bridge configuration
```

```
*
*talk 5

+protocol asrt
ASRT>list bridge
SRB Bridge ID (prio/add):  32768/02-00-41-08-00-80
Bridge state:          Enabled
UB-Encapsulation:      Disabled
Bridge type:           SRB
Number of ports:       2
STP Participation:     IBM-SRB proprietary

                                                       Maximum
Port  Interface      State  MAC Address        Modes   MSDU  Segment   Flags
   1  TKR/0            Up   02-00-41-08-00-80    SR      4544  001       RD    1
   2  TKR/1            Up   02-00-41-08-00-40    SR      4544  002       RD    1

Flags:  RE = IBMRT PC behavior Enabled,  RD = IBMRT PC behavior Disabled

SR bridge number:      1
SR virtual segment:    000
Adaptive segment:      000
ASRT>flip                2
MAC address [00-00-00-00-00-00]? 02-00-41-08-00-80
IEEE 802 canonical bit order:    02-00-41-08-00-80
IBM Token-Ring native bit order: 40:00:82:10:00:01
ASRT>flip                2
MAC address [00-00-00-00-00-00]? 02-00-41-08-00-40
IEEE 802 canonical bit order:    02-00-41-08-00-40
IBM Token-Ring native bit order: 40:00:82:10:00:02
ASRT>exit
+

*
```

Note:

1 Both port/interfaces are active

2 Using flip to confirm the interfaces MAC addresses

Figure 303 (Part 2 of 2). Restart and Verify

11.4.17 IBM 8210 Transparent Bridging

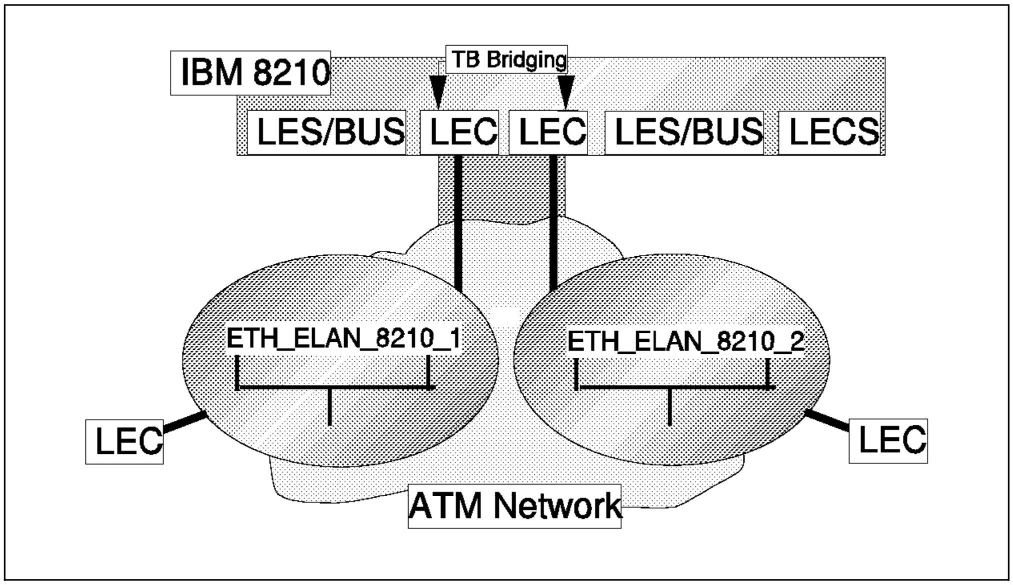


Figure 304. Transparent Bridging

Figure 304 depicts a configuration where the IBM 8210 Nways MSS Server is connected to two Ethernet ELANs and is enabled for TB bridging between them.

Both ELANs are controlled by local LES/BUSs. An LECS has been defined to enable LE clients to join the ELANs. The LECS assigns LE clients to the ELANs using a name policy. No security is enforced, and LE clients can join without connecting to the LECS first.

To enable IP access to the MSS Server′s management functions we have configured the IP host services.

Although not critical for the bridging function, we have enabled BCM for IP, IPX, and NetBIOS on both Ethernet ELANs.

To realize the scenario depicted in Figure 304 we used the parameters depicted in Figure 305 on page 398. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 306. Configure ATM Interface

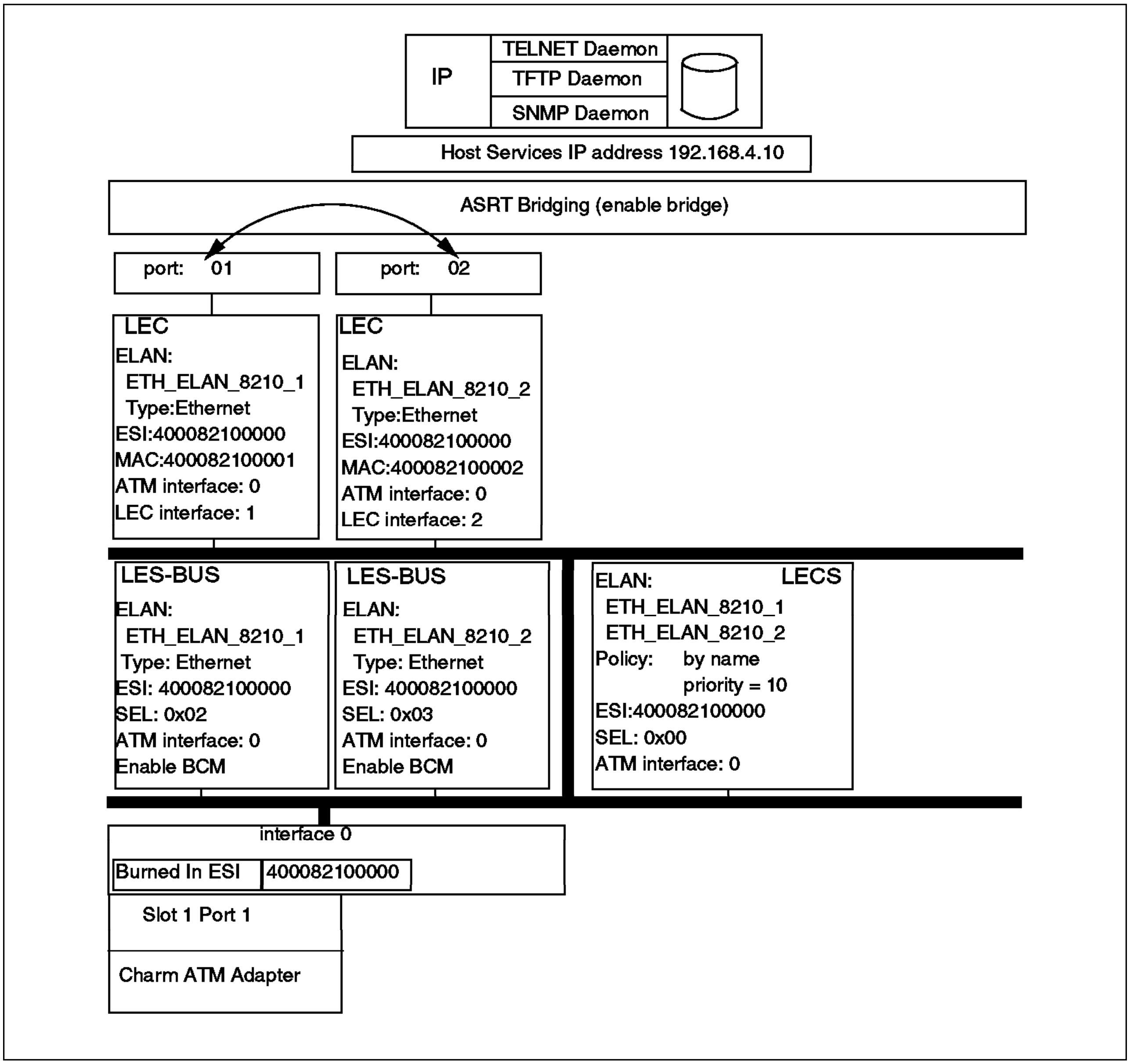


Figure 305. Transparent Bridging - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 **Configure the LECS**

Add the LECS

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 307 (Part 1 of 6). Configure LECS

Add ELAN ′ETH_ELAN_8210_1′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? ETH_ELAN_8210_1
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 1
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 1
ELAN ′ETH_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 307 (Part 2 of 6). Configure LECS

Add ELAN ′ETH_ELAN_8210_2′ to the LECS

```
LECS ELANs config>
LECS ELANs config>add
Name of ELAN []? ETH_ELAN_8210_2
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 1
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 1

ELAN ′ETH_ELAN_8210_2′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 307 (Part 3 of 6). Configure LECS

Configure the ′ETH_ELAN_8210_1′ ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_1
  (  2) ETH_ELAN_8210_2
Choice of ELAN [1]? 1
ELAN ′ETH_ELAN_8210_1′ selected for detailed configuration
Selected ELAN ′ETH_ELAN_8210_1′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: ETH_ELAN_8210_1
    added to ELAN ′ETH_ELAN_8210_1′
Selected ELAN ′ETH_ELAN_8210_1′>
Selected ELAN ′ETH_ELAN_8210_1′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: ETH_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? ETH_ELAN_8210_1
ELAN name ′ETH_ELAN_8210_1′
     bound to LES Local LES for: ETH_ELAN_8210_1
Selection ″ELAN name add″ Complete
Selected ELAN ′ETH_ELAN_8210_1′>exit
LECS ELANs config>
```

Figure 307 (Part 4 of 6). Configure LECS

Configure the ′ETH_ELAN_8210_2′ ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_1
  (  2) ETH_ELAN_8210_2
Choice of ELAN [1]? 2
ELAN ′ETH_ELAN_8210_2′ selected for detailed configuration
Selected ELAN ′ETH_ELAN_8210_2′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: ETH_ELAN_8210_2
    added to ELAN ′ETH_ELAN_8210_2′
Selected ELAN ′ETH_ELAN_8210_2′>
Selected ELAN ′ETH_ELAN_8210_2′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: ETH_ELAN_8210_2


Enter Selection:  [1]? 1
ELAN name []? ETH_ELAN_8210_2                      .
ELAN name ′ETH_ELAN_8210_2′
     bound to LES Local LES for: ETH_ELAN_8210_2
Selection ″ELAN name add″ Complete
Selected ELAN ′ETH_ELAN_8210_2′>exit
LECS ELANs config>exit
LECS config>
```

Figure 307 (Part 5 of 6). Configure LECS

Enable LECS policies

```
LECS config>
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete

LECS POLICIES config>list all

Policy Listing...

Enabled  Priority  Type
=======  ========  ==========
    Yes        10  byElanNm
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 307 (Part 6 of 6). Configure LECS

3 Configure the LES/BUS

```
Add the LES-BUS to the 'ETH_ELAN_8210_1' ELAN
```

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN 'ETH_ELAN_8210_1'>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 2
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2


Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'ETH_ELAN_8210_1'>enable bcm all   1
LES-BUS config for ELAN 'ETH_ELAN_8210_1'>exit
LE Services config>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

Figure 308 (Part 1 of 2). Configure LES/BUS

Add the LES-BUS to the ′ETH_ELAN_8210_2′ ELAN

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_2
LES-BUS configuration
LES-BUS config for ELAN ′ETH_ELAN_8210_2′>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2



Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [3]? 3
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′ETH_ELAN_8210_2′>enable bcm all   1
LES-BUS config for ELAN ′ETH_ELAN_8210_2′>exit
LE Services config>exit
ATM Config>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

Figure 308 (Part 2 of 2). Configure LES/BUS

4 Configure the LEC

Configure an LEC for the ′ETH_ELAN_8210_1′ ELAN

```
ATM Config>
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8210_1
Ethernet Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x4
Ethernet Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Ethernet Forum Compliant LEC Config>
Ethernet Forum Compliant LEC Config>exit
LE Client config>
```

Figure 309 (Part 1 of 2). Configure LEC

Configure an LEC for the ′ETH_ELAN_8210_2′ ELAN

```
LE Client config>
LE Client config>add ethernet
Added Emulated LAN as interface 2
LE Client config>config
Emulated LAN interface number [1]? 2
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8210_2
Ethernet Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x5
Ethernet Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.02
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 309 (Part 2 of 2). Configure LEC

5 Configure the transparent bridge

Configure the transparent bridge

```
Config>
Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>enable bridge
ASRT config>list port
Port ID (dec)    : 128:01, (hex): 80-01            1
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only       2
Assoc Interface  : 1
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:02, (hex): 80-02            1
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only       2
Assoc Interface  : 2
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Note:

1 These are the port numbers assigned to each interface and used for the bridge configuration.

2 By default transparent bridging is enable when you enable bridge

Figure 310 (Part 1 of 2). Configure Transparent Bridge

List bridge

```
ASRT config>list bridge

                Source Routing Transparent Bridge Configuration
                ===============================================

Bridge:                  Enabled                  Bridge Behavior: STB
                   +----------------------------+
-------------------| SOURCE ROUTING INFORMATION |-------------------------------

                   +----------------------------+
Bridge Number:           N/A                      Segments:          0
Max ARE Hop Cnt:         00                       Max STE Hop cnt:   00
1:N SRB:                 Not Active               Internal Segment:  0x000
LF-bit interpret:        Extended
                   +-------------------+
-------------------| SR-TB INFORMATION |----------------------------------------

                   +-------------------+
SR-TB Conversion:        Disabled
TB-Virtual Segment:      0x000                    MTU of TB-Domain:  0
                   +------------------------------------+
-------------------| SPANNING TREE PROTOCOL INFORMATION |-----------------------

                   +------------------------------------+
Bridge Address:          Default                  Bridge Priority:   32768/0x8000
STP Participation:       IEEE802.1d
                   +-------------------------+
-------------------| TRANSLATION INFORMATION |----------------------------------

                   +-------------------------+
FA<=>GA Conversion:      Enabled                  UB-Encapsulation:  Disabled
                   +------------------+
-------------------| PORT INFORMATION |-----------------------------------------

                   +------------------+
Number of ports added: 2
Port:   1      Interface:      1      Behavior:    STB Only   STP:  Enabled   1
Port:   2      Interface:      2      Behavior:    STB Only   STP:  Enabled   1

ASRT config>exit
Config>
```

Note:

1 STB is enabled on both interfaces

Figure 310 (Part 2 of 2). Configure Transparent Bridge

6 Configure IP host services

```
Config>
Config>protocol hst
TCP/IP-Host Services user configuration
TCP/IP-Host config>set ip-host address
IP-Host address [0.0.0.0]? 192.168.4.10       1
Address mask [255.255.255.0]? 255.255.255.0
IP-Host address set.

TCP/IP-Host config>list all

IP-Host IP address : 192.168.4.10
Address Mask       : 255.255.255.0

No Default Gateway address currently configured.

TCP/IP-Host Services Enabled.

RIP-LISTENING Disabled.

Router Discovery Enabled.

TCP/IP-Host config>exit
Config>
```

Note:

1 This IP address will be used for management only

Figure 311. Configure IP Host Services

7 Restart the MSS Server and verify the configuration

Restart the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 312 (Part 1 of 2). Restart and Verify

Verify the bridge configuration

```
*
*talk 5

+protocol asrt
ASRT>list bridge
SRB Bridge ID (prio/add):  32768/40-00-82-10-00-01
Bridge state:          Enabled
UB-Encapsulation:      Disabled
Bridge type:           STB
Number of ports:       2
STP Participation:     IEEE802.1d

                                                      Maximum
Port  Interface      State  MAC Address        Modes   MSDU   Segments  Flags
   1  Eth/0           Up    40-00-82-10-00-01       T    1516            RD    1
   2  Eth/1           Up    40-00-82-10-00-02       T    1516            RD    1

Flags:  RE = IBMRT PC behavior Enabled,  RD = IBMRT PC behavior Disabled

SR bridge number:      1
SR virtual segment:    000
Adaptive segment:      000
ASRT>exit
+

*
```

Note:

1 Both port/interfaces are active

Figure 312 (Part 2 of 2). Restart and Verify

11.4.18 IBM 8210 Source Route Translational Bridging (SR-TB)

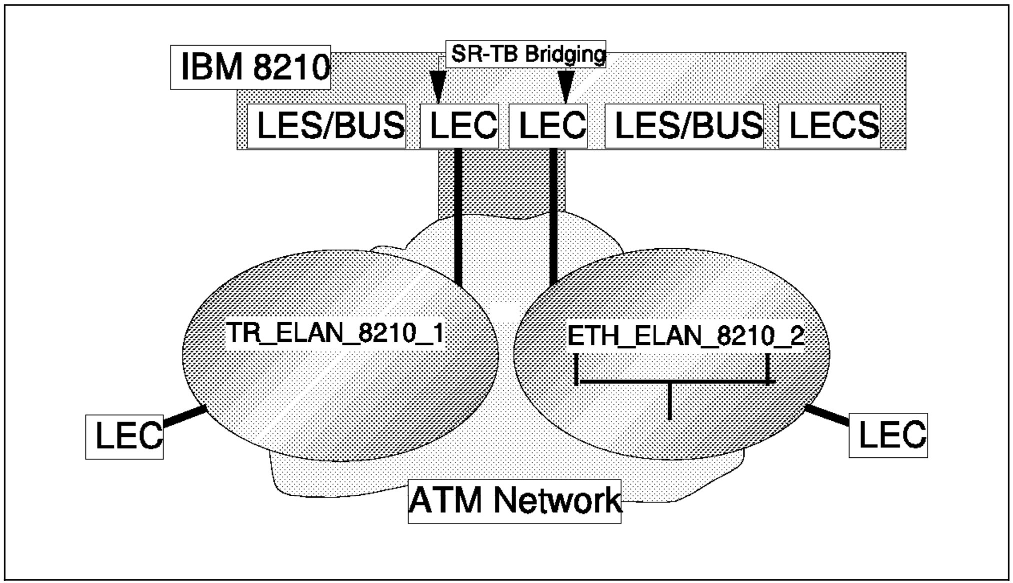


Figure 313. Source Route Translational Bridging (SR-TB)

Figure 313 depicts a configuration where the IBM 8210 Nways MSS Server is connected to two ELANs, token-ring and Ethernet respectively, and is enabled for translational (SR-TB) bridging between them.

Note: The bridging functions are for NetBIOS and IEEE 802.2 LLC only (for example, SNA). IP or IPX routing has to be enabled to provide IP or IPX connectivity between the ELANs.

Both ELANs are controlled by local LES/BUSs. An LECS has been defined to enable LE clients to join the ELANs. The LECS assigns LE clients to the ELANs using a name policy. No security is enforced, and LE clients can join without connecting to the LECS first.

To enable IP access to the MSS Server′s management functions we have configured the IP host services.

Although not critical for the bridging function, we have enabled BCM for IP, IPX, and NetBIOS on both ELANs. SRM has been enabled on the token-ring ELAN.

To realize the scenario depicted in Figure 313 we used the parameters depicted in Figure 314 on page 411. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface**

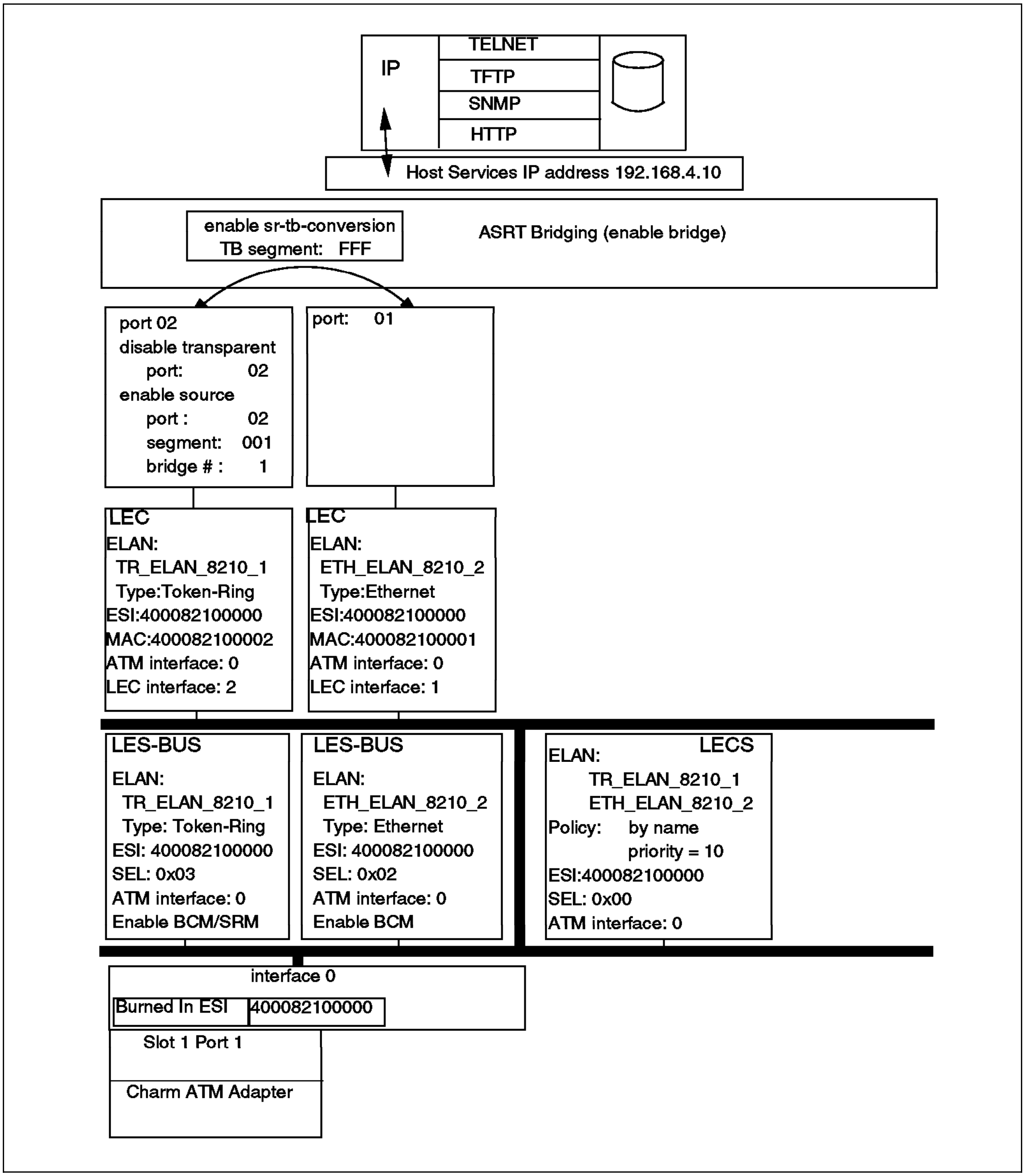


Figure 314. Translational Bridging - Parameter Overview

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 315. Configure ATM Interface

> **Important**
>
> Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 Configure the LECS

Add the LECS

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.00
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 316 (Part 1 of 6). Configure LECS

Add ELAN ′ETH_ELAN_8210_2′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? ETH_ELAN_8210_2
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 1
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 1
ELAN ′ETH_ELAN_8210_2′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 316 (Part 2 of 6). Configure LECS

Add ELAN ′TR_ELAN_8210_1′ to the LECS

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
Type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 316 (Part 3 of 6). Configure LECS

Configure the ′ETH_ELAN_8210_2′ ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_2
  (  2) TR_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN ′ETH_ELAN_8210_2′ selected for detailed configuration
Selected ELAN ′ETH_ELAN_8210_2′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: ETH_ELAN_8210_2
    added to ELAN ′ETH_ELAN_8210_2′
Selected ELAN ′ETH_ELAN_8210_2′>
Selected ELAN ′ETH_ELAN_8210_2′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: ETH_ELAN_8210_2

Enter Selection:  [1]? 1
ELAN name []? ETH_ELAN_8210_2
ELAN name ′ETH_ELAN_8210_2′
     bound to LES Local LES for: ETH_ELAN_8210_2
Selection ″ELAN name add″ Complete
Selected ELAN ′ETH_ELAN_8210_2′>exit
LECS ELANs config>
```

Figure 316 (Part 4 of 6). Configure LECS

Configure the ′TR_ELAN_8210_1′ ELAN, and set policy value

```
LECS ELANs config>
LECS ELANs config>select
  (  1) ETH_ELAN_8210_2
  (  2) TR_ELAN_8210_1
Choice of ELAN [1]? 2
ELAN ′TR_ELAN_8210_1′ selected for detailed configuration
Selected ELAN ′TR_ELAN_8210_1′>less add
  (  1) Local
  (  2) Remote
Primary LES is [1]? 1
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is  [1]? 1
LES ATM address Local LES for: TR_ELAN_8210_1
    added to ELAN ′TR_ELAN_8210_1′
Selected ELAN ′TR_ELAN_8210_1′>
Selected ELAN ′TR_ELAN_8210_1′>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_1


Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1                      .
ELAN name ′TR_ELAN_8210_1′
     bound to LES Local LES for: TR_ELAN_8210_1
Selection ″ELAN name add″ Complete
Selected ELAN ′TR_ELAN_8210_1′>exit
LECS ELANs config>exit
LECS config>
```

Figure 316 (Part 5 of 6). Configure LECS

Enable LECS policies

```
LECS config>
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete

LECS POLICIES config>list all

Policy Listing...

Enabled  Priority  Type
=======  ========  ==========
    Yes        10  byElanNm
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 316 (Part 6 of 6). Configure LECS

3 Configure the LES/BUS

Add the LES/BUS for the ′ETH_ELAN_8210_2′ ELAN

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_2
LES-BUS configuration
LES-BUS config for ELAN ′ETH_ELAN_8210_2′>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 2
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′ETH_ELAN_8210_2′>enable bcm all   1
LES-BUS config for ELAN ′ETH_ELAN_8210_2′>exit
LE Services config>
```

Figure 317 (Part 1 of 2). Configure LES/BUS

```
Add the LES/BUS for the 'TR_ELAN_8210_1' ELAN
```

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN 'TR_ELAN_8210_1'>add
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2



Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [3]? 3
Selection "Add LES-BUS" Complete
LES-BUS config for ELAN 'TR_ELAN_8210_1'>enable bcm all   1
LES-BUS config for ELAN 'TR_ELAN_8210_1'>enable source    2
LES-BUS config for ELAN 'TR_ELAN_8210_1'>exit
LE Services config>exit
ATM Config>
```

Note:

1 Enabling Broadcast Management for NetBIOS, IP and IPX

2 Enabling Source Routing Management for NetBIOS, IP and IPX

Figure 317 (Part 2 of 2). Configure LES/BUS

4 Configure the LEC

Configure an LEC for the ′ETH_ELAN_8210_2′ ELAN

```
ATM Config>
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? ETH_ELAN_8210_2
Ethernet Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x4
Ethernet Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Ethernet Forum Compliant LEC Config>
Ethernet Forum Compliant LEC Config>exit
LE Client config>
```

Figure 318 (Part 1 of 2). Configure LEC

Configure an LEC for the ′TR_ELAN_8210_1′ ELAN

```
LE Client config>
LE Client config>add token
Added Emulated LAN as interface 2
LE Client config>config
Emulated LAN interface number [1]? 2
ATM LAN Emulation Client configuration
Token Ring Forum Compliant LEC Config>set elan-name
Assign emulated LAN name []? TR_ELAN_8210_1
Token Ring Forum Compliant LEC Config>set esi
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x5
Token Ring Forum Compliant LEC Config>set mac
Use adapter address for MAC? [Yes]: no
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.02
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 318 (Part 2 of 2). Configure LEC

5 **Configure the translational bridge**

Configure the translational bridge

```
Config>
Config>protocol asrt
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>enable bridge
ASRT config>list port
Port ID (dec)    : 128:01, (hex): 80-01            1
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only       2
Assoc Interface  : 1
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:02, (hex): 80-02            1
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only       2
Assoc Interface  : 2
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


ASRT config>disable transparent 2        3
ASRT config>enable source-routing 2      4
Segment Number for the port in hex(1 - FFF) [  1]? 001
Bridge number in hex (0 - 9, A - F) [0]? 1
ASRT config>enable sr-tb-conversion      5
TB-Domain Segment Number in hex(1 - FFF) [1]? FFF   6
TB-Domain's MTU [1470]? 1470
```

Note:

1 These are the port numbers assigned to each interface and used for the bridge configuration.

2 By default transparent bridging is enable when you enable bridge.

3 Disabling transparent bridging in the token ring port.

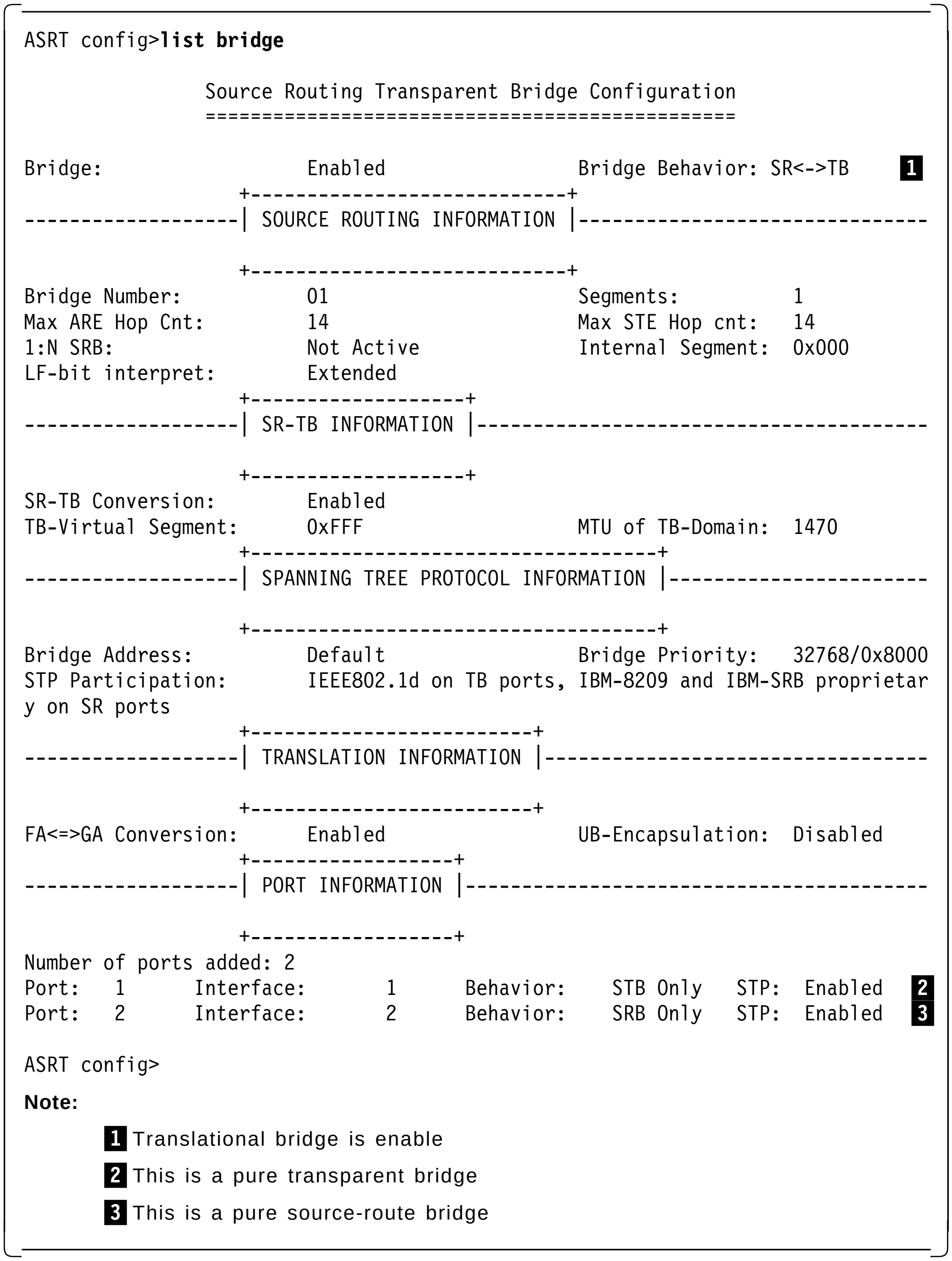
4 Enabling source routing bridging in the token ring port.

5 Enabling translational bridging.

6 This is the virtual segment number for the translational bridge.

Figure 319 (Part 1 of 2). Configure Translational Bridge

List bridge



```
ASRT config>list bridge

                Source Routing Transparent Bridge Configuration
                ===============================================

Bridge:                  Enabled                 Bridge Behavior: SR<->TB     1
                   +----------------------------+
-------------------| SOURCE ROUTING INFORMATION |-------------------------------

                   +----------------------------+
Bridge Number:           01                      Segments:          1
Max ARE Hop Cnt:         14                      Max STE Hop cnt:   14
1:N SRB:                 Not Active              Internal Segment:  0x000
LF-bit interpret:        Extended
                   +-------------------+
-------------------| SR-TB INFORMATION |----------------------------------------

                   +-------------------+
SR-TB Conversion:        Enabled
TB-Virtual Segment:      0xFFF                   MTU of TB-Domain:  1470
                   +------------------------------------+
-------------------| SPANNING TREE PROTOCOL INFORMATION |-----------------------

                   +------------------------------------+
Bridge Address:          Default                 Bridge Priority:   32768/0x8000
STP Participation:       IEEE802.1d on TB ports, IBM-8209 and IBM-SRB proprietar
y on SR ports
                   +-------------------------+
-------------------| TRANSLATION INFORMATION |----------------------------------

                   +-------------------------+
FA<=>GA Conversion:      Enabled                 UB-Encapsulation:  Disabled
                   +------------------+
-------------------| PORT INFORMATION |-----------------------------------------

                   +------------------+
Number of ports added: 2
Port:   1      Interface:      1      Behavior:    STB Only   STP:  Enabled    2
Port:   2      Interface:      2      Behavior:    SRB Only   STP:  Enabled    3

ASRT config>
```

Note:

1 Translational bridge is enable

2 This is a pure transparent bridge

3 This is a pure source-route bridge

Figure 319 (Part 2 of 2). Configure Translational Bridge

6 Configure IP host services

```
Config>
Config>protocol hst
TCP/IP-Host Services user configuration
TCP/IP-Host config>set ip-host address
IP-Host address [0.0.0.0]? 192.168.4.10        1
Address mask [255.255.255.0]? 255.255.255.0
IP-Host address set.

TCP/IP-Host config>list all

IP-Host IP address : 192.168.4.10
Address Mask       : 255.255.255.0

No Default Gateway address currently configured.

TCP/IP-Host Services Enabled.

RIP-LISTENING Disabled.

Router Discovery Enabled.

TCP/IP-Host config>exit
Config>
```

Note:

1 This IP address will be used for management only

Figure 320. Configure IP Host Services

7 Restart the MSS Server and verify the configuration

Restarting the MSS Server to activate the new configuration

```
Config>   <Ctrl+P>

*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 321 (Part 1 of 2). Restart and Verify

Verify the bridge configuration

```
*talk 5

+
+protocol asrt
ASRT>list bridge
Bridge ID (prio/add):      32768/40-00-82-10-00-01
Bridge state:          Enabled
UB-Encapsulation:      Disabled
Bridge type:           SR-TB
Number of ports:       2
STP Participation:     IEEE802.1d on TB ports and IBM-8209 on SR ports

                                                       Maximum
Port  Interface      State  MAC Address        Modes   MSDU   Segment   Flags
   1  Eth/0            Up   40-00-82-10-00-01      T    1516            RD    1
   2  TKR/0            Up   02-00-41-08-00-40    SR     4544  001       RD    1

Flags:  RE = IBMRT PC behavior Enabled,  RD = IBMRT PC behavior Disabled

SR bridge number:      1
SR virtual segment:    000
Adaptive segment:      FFF
ASRT>flip                 2
MAC address [00-00-00-00-00-00]? 02-00-41-08-00-40
IEEE 802 canonical bit order:    02-00-41-08-00-40
IBM Token-Ring native bit order: 40:00:82:10:00:02
ASRT>exit
+

*
```

Note:

1 Both port/interfaces are active

2 Using flip to confirm the interface MAC address

Figure 321 (Part 2 of 2). Restart and Verify

11.4.19 SNMP Functions

This section describes how to configure the 8210 to enable a group of workstation to read and write the SNMP variables, and to send SNMP traps to a specific network management stations.

The scenario that need to be set up:

- Community name: public
- Access to: all SNMP variables
- Read/write access for:
  - All hosts within subnet 9.24.104
  - All hosts within subnet 192.186
- SNMP traps to 9.24.104.110
- All traps will be sent

Note: This scenario assumes that IP functions have already been configured on the 8210.

The definitions required to achieve this scenario are:

Add community public

```
*talk 6
Config>protocol SNMP
SNMP Config>add community public
Community added successfully
```

Figure 322 (Part 1 of 5). Define and Display SNMP Definitions

Define access type

```
SNMP Config>set community access write_read_trap
Community name
[]? public
Access set successfully
```

Figure 322 (Part 2 of 5). Define and Display SNMP Definitions

Define hosts addresses within community public

```
SNMP Config>add address public
IP Address [0.0.0.0]? 192.168.0.0
IP Mask [255.255.255.255]? 255.255.0.0
Address added successfully
SNMP Config>add address public
IP Address [0.0.0.0]? 9.24.104.0
IP Mask [255.255.255.255]? 255.255.255.0
Address added successfully
SNMP Config>add address public
IP Address [0.0.0.0]? 9.24.104.110
IP Mask [255.255.255.255]?
Address added successfully
```

Figure 322 (Part 3 of 5). Define and Display SNMP Definitions

Enable all traps

```
SNMP Config>enable trap all
Community name []? public
Trap(s) enabled successfully
```

Figure 322 (Part 4 of 5). Define and Display SNMP Definitions

Display SNMP configuration

```
SNMP Config>list all

SNMP is enabled
Trap UDP port: 162

          Community Name                 Access
-------------------------------  -------------------
public                            Read, Write, Trap



         Community Name             IP Address        IP Mask
-------------------------------  ---------------  ---------------
public                            192.168.0.0      255.255.0.0
                                  9.24.104.0       255.255.255.0
                                  9.24.104.110     255.255.255.255



         Community Name                   Enabled Traps
-------------------------------  --------------------------------
public                            Cold Restart, Warm Restart,
                                  Link Down, Link Up,
                                  Authentication Failure,
                                  EGP Neighbor Loss, Enterprise Specific



          Community Name                       View
-------------------------------  --------------------------------
public                            All


There are no views

SNMP Config>exit
Config>
```

Figure 322 (Part 5 of 5). Define and Display SNMP Definitions

11.4.20 LECS and LES/BUS Redundancy

This section describes how to configure two 8210s to provide mutual backup for LES/BUS and LECS functions.

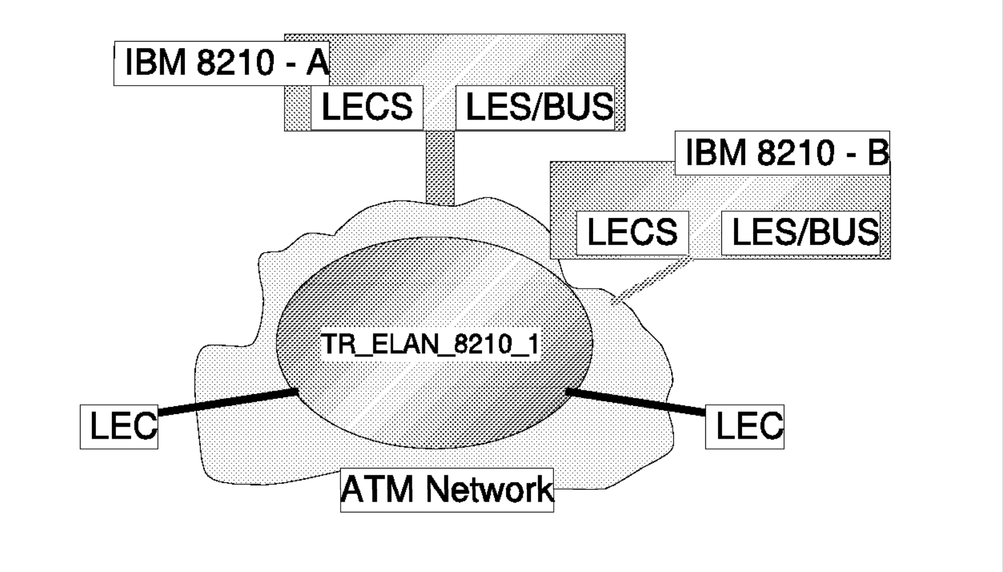


Figure 323. Multiple LECSs and LES/BUSs

Figure 323 shows a single token-ring ELAN with two 8210s (8210-A and 8210-B). 8210-A is configured as the primary LES/BUS for the ELAN, while 8210-B is configured as the backup LES/BUS. By enabling the ELAN redundancy feature on both 8210s, a redundancy VCC will be established between both LESs.

On both 8210s identical LECS functions have been configured. The LES on 8210-A is configured as the primary, and the LES on 8210-B is configured as the backup server. LECS backup is provided by defining both ATM addresses in the ATM switches to which any of the LE clients connect.

Note: This backup scenario assumes that LE clients learn their LES address from the LECS, while they learn the LECS address for their adjacent ATM switch (see 5.5, "Redundant LECS" on page 108).

To realize the scenario depicted in Figure 323 we used the parameters depicted in Figure 324 on page 426. For the parameters not shown we used default values. The configuration steps required are:

1 **Configure the ATM interface on 8210-A**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.0A
ATM Interface Config>exit
ATM Config>
```

Figure 325. Configure ATM Interface

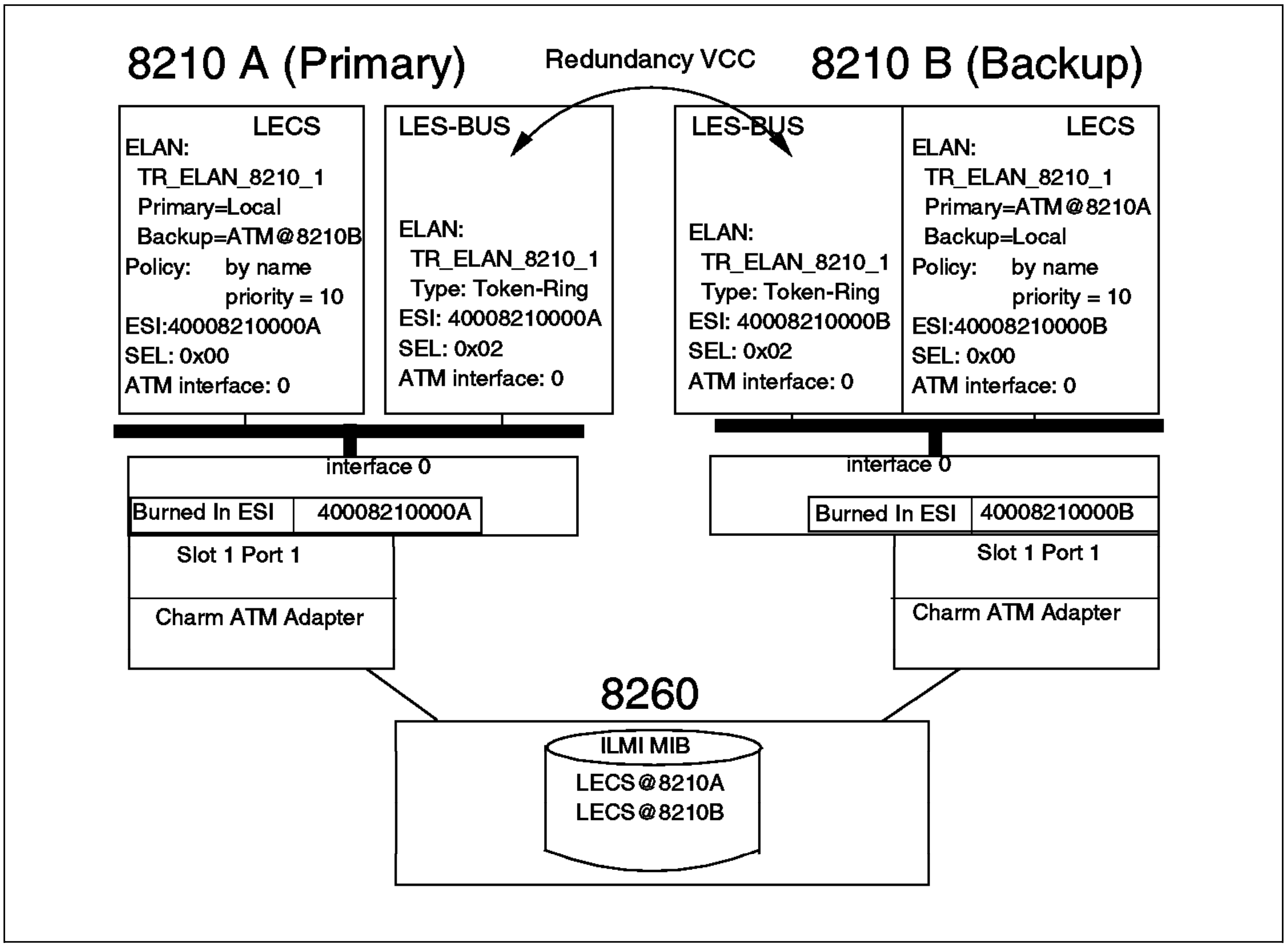


Figure 324. Redundant LES/BUS and LECS - Parameter Overview

Important

Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

2 Configure LECS on 8210-A

Add the LECS on 8210-A

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.0A
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 326 (Part 1 of 4). Configure LECS on 8210-A

Add ELAN ′TR_ELAN_8210_1′ to the LECS on 8210-A

```
LECS config>
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 326 (Part 2 of 4). Configure LECS on 8210-A

Configure the TR_ELAN_8210_1 ELAN on 8210-A

```
LECS ELANs config>
LECS ELANs config>select
  (1) TR_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN 'TR_ELAN_8201_1' selected for detailed configuration
Selected ELAN 'TR_ELAN_8210_1'>less add
  (1) Local
  (2) Remote
Primary LES is [1]? 1
  (1) Unspecified
  (2) Local
  (3) Remote
Backup LES is  [1]? 3
If backup LES is remote, enter ATM address []?
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.0B.02  1
LES ATM address Local LES for: TR_ELAN_8210_1
    added to ELAN 'TR_ELAN_8210_1'
Selected ELAN 'TR_ELAN_8210_1'>list all
ELAN Configuration:
  ELAN is         Enabled
  Name:           'TR_ELAN_8210_1'
  ELAN Type:      TokenRing
  Max Frame Size:  4544
Selected ELAN 'TR_ELAN_8210_1'>
Selected ELAN 'TR_ELAN_8210_1'>policy add name
ATM address of LES for policy value(s)

        (1) Local LES for: TR_ELAN_8210_1

Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1
ELAN name 'TR_ELAN_8210_1'
     bound to LES Local LES for: TR_ELAN_8210_1
Selection "ELAN name add" Complete
Selected ELAN 'TR_ELAN_8210_1'>exit
LECS ELANs config>exit
LECS config>
```

Note:

1 This is the ATM address of 8210-B's LES/BUS

Figure 326 (Part 3 of 4). Configure LECS on 8210-A

Enable LECS policies on 8210-A

```
LECS config>
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 326 (Part 4 of 4). Configure LECS on 8210-A

3 Configure the LES/BUS on 8210-A

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN ′TR_ELAN_8210_1′>add
Turn on Standard Event Logging for LES [yes]
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.0A

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′TR_ELAN_8210_1′>
```

Figure 327. Configure LES/BUS on 8210-A

4 Enable LES/BUS redundancy on 8210-A

```
LES-BUS config for ELAN ′TR_ELAN_8210_1′>enable redundancy
Redundancy protocol role
        (1) Primary LES-BUS
        (2) Backup LES-BUS

Enter Selection:  [1]? 1
ATM address of backup les-bus  [1]?
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.0B.02     1
Selection ″Enable Redundancy″ Complete
LES-BUS config for ELAN ′TR_ELAN_8210_1′>list all
LES-BUS Detailed Configuration
Name:  TR_ELAN_8210_1
   LES-BUS Enabled/Disabled:                  Enabled
   ATM Device number:                         0
   End System Identifier (ESI):               40.00.82.10.00.0A
   Selector Byte:                             0x02
   ELAN Type:       (S2)                      Token Ring
   Max Frame Size:  (S3)                      4544
   Control Timeout: (S4)                      120
   Max Frame Age:   (S5)                      1
   Validate Best Effort Peak Cell Rate (PCR): No
   Control Distribute VCC Traffic Type:       Best Effort VCC
   Control Distribute VCC PCR in Kbps:        155000
   Control Direct VCC Max Reserved Bandwidth: 0
   Multicast Forward VCC Traffic Type:        Best Effort VCC
   Multicast Forward VCC PCR in Kbps:         155000
   Multicast Send VCC MAX Reserved Bandwidth: 0

    -LES-BUS Options-
   Security (LECS Validation of Joins):       Disabled
   Partition LE_ARP_REQUEST Forwarding Domain: Yes
   LE_ARP RESPONSE Destination:               One client
   Partition Unicast Frame Domain:            Yes
   Redundancy:                                Enabled       2
   Redundancy Role:                           Primary LES-BUS   3
   ATM address of Backup LES-BUS:  3909851111111111111111010140008210000B02
   ATM address trace filter value: 0000000000000000000000000000000000000000
   ATM address trace filter mask:  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    -BUS Monitor Configuration-
   Monitor Host Usage of BUS:                 Disabled
   # Top Hosts to Record:                     10
   # Seconds in each sample interval:         10
   # Minutes between sample intervals:        30
   Frame sampling rate:                       1 out of 10

    -Broadcast Manager Configuration-
   IP BCM:                                    Disabled
   IPX BCM:                                   Disabled
   NetBIOS BCM:                               Disabled
   BCM IP Cache Aging Time:                   5
   BCM IPX Cache Aging Time:                  3
   BCM NetBIOS Cache Aging Time:              15
   Token Ring Source Route Management:        Disabled
   No BCM Static Entries defined
```

Note:

1 This is the ATM address of 8210-B′s LES/BUS

2 Redundancy option is enabled on 8210-A

3 8210-A is configured in a primary LES/BUS mode

Figure 328. Enabling LES/BUS Redundancy on 8210-A

5 Configure the ATM interface on 8210-B

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.0B
ATM Interface Config>exit
ATM Config>
```

Figure 329. Configure ATM Interface

> **Important**
>
> Make sure the LECS address is defined on the adjacent ATM switch (see 2 on page 296).

6 Configure the LECS on 8210-B

Add the LECS on 8210-B

```
ATM Config>
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>lecs
Lan Emulation Configuration Server configuration
LECS config>add
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.0B
End system identifier [1]? 2
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:yes
Standard ELS activated for LECS
LECS config>
```

Figure 330 (Part 1 of 4). Configure LECS on 8210-B

Add ELAN ′TR_ELAN_8210_1′ to the LECS on 8210-B

```
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>add
Name of ELAN []? TR_ELAN_8210_1
type of ELAN
        (1) Ethernet
        (2) TokenRing

Enter Selection:  [2]? 2
Maximum frame size of ELAN
        (1) 1516
        (2) 4544
        (3) 9234
        (4) 18190

Enter Selection:  [2]? 2
ELAN ′TR_ELAN_8210_1′ added
Selection ″ELAN addition″ Complete
LECS ELANs config>
```

Figure 330 (Part 2 of 4). Configure LECS on 8210-B

Configure the ’TR_ELAN_8210_1’ ELAN on 8210-B

```
LECS ELANs config>select
  (1) TR_ELAN_8210_1
Choice of ELAN [1]? 1
ELAN ’TR_ELAN_8201_1’ selected for detailed configuration
Selected ELAN ’TR_ELAN_8210_1’>less add
  (1) Local
  (2) Remote
Primary LES is [1] 2
If primary LES is remote, enter ATM address []?
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.0A.02
  (  1) Unspecified
  (  2) Local
  (  3) Remote
Backup LES is [2]
LES ATM address 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.0A.02
    added to ELAN ’TR_ELAN_8210_1’
Selected ELAN ’TR_ELAN_8210_1’>list
ELAN Configuration:
  ELAN is         Enabled
  Name:           ’TR_ELAN_8210_1’
  ELAN Type:      TokenRing
  Max Frame Size:  4544

Selected ELAN ’TR_ELAN_8210_1’>less list


LESs for ELAN ’TR_ELAN_8210_1’

Enbld  LES ATM address
=====  ===========================================================
  Yes  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.0A.02
       backup: Local LES for: TR_ELAN_8210_1
Selected ELAN ’TR_ELAN_8210_1’>
Selected ELAN ’TR_ELAN_8210_1’>
Selected ELAN ’TR_ELAN_8210_1’>policy add name
ATM address of LES for policy value(s)

       (1) 39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.0A.02

Enter Selection:  [1]? 1
ELAN name []? TR_ELAN_8210_1
ELAN name ’TR_ELAN_8210_1’
     bound to LES 39.09.85.11.11.11.11.11.11.11.11.01.01.
                  40.00.82.10.00.0A.02
Selection ″ELAN name add″ Complete
Selected ELAN ’TR_ELAN_8210_1’>exit
LECS ELANs config>exit
LECS config>
```

Figure 330 (Part 3 of 4). Configure LECS on 8210-B

Enable LECS policies on 8210-B

```
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>add
Priority of Policy [10]? 10
Policy type
        (1) byAtmAddr
        (2) byMacAddr
        (3) byRteDesc
        (4) byLanType
        (5) byPktSize
        (6) byElanNm

Enter Selection:  [1]? 6
Added policy ′byElanNm ′ at priority 10
Selection ″Add assignment policy″ Complete
LECS POLICIES config>exit
LECS config>exit
LE Services config>
```

Figure 330 (Part 4 of 4). Configure LECS on 8210-B

7 Configure the LES/BUS on 8210-B

```
LE Services config>
LE Services config>les-bus
ELAN Name (ELANxx) []? TR_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN ′TR_ELAN_8210_1′>add
Turn on Standard Event Logging for LES [yes]
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 1
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.0B

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′TR_ELAN_8210_1′>
```

Figure 331. Configure LES/BUS on 8210-B

8 Enable LES/BUS Redundancy on 8210-B

```
LES-BUS config for ELAN ’TR_ELAN_8210_1’>enable redundancy
Redundancy protocol role
        (1) Primary LES-BUS
        (2) Backup LES-BUS

Enter Selection:  [1]? 2
Selection ″Enable Redundancy″ Complete
LES-BUS config for ELAN ’TR_ELAN_8210_1’>list all
LES-BUS Detailed Configuration
Name:  TR_ELAN_8210_1
   LES-BUS Enabled/Disabled:                   Enabled
   ATM Device number:                          0
   End System Identifier (ESI):                40.00.82.10.00.0B
   Selector Byte:                              0x02
   ELAN Type:       (S2)                       Token Ring
   Max Frame Size:  (S3)                       4544
   Control Timeout: (S4)                       120
   Max Frame Age:   (S5)                       1
   Validate Best Effort Peak Cell Rate (PCR):  No
   Control Distribute VCC Traffic Type:        Best Effort VCC
   Control Distribute VCC PCR in Kbps:         155000
   Control Direct VCC Max Reserved Bandwidth:  0
   Multicast Forward VCC Traffic Type:         Best Effort VCC
   Multicast Forward VCC PCR in Kbps:          155000
   Multicast Send VCC MAX Reserved Bandwidth:  0

    -LES-BUS Options-
   Security (LECS Validation of Joins):        Disabled
   Partition LE_ARP_REQUEST Forwarding Domain: Yes
   LE_ARP RESPONSE Destination:                One client
   Partition Unicast Frame Domain:             Yes
   Redundancy:                                 Enabled          1
   Redundancy Role:                            Backup LES-BUS   2
   ATM address trace filter value: 0000000000000000000000000000000000000000
   ATM address trace filter mask:  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
    -BUS Monitor Configuration-
   Monitor Host Usage of BUS:                  Disabled
   # Top Hosts to Record:                      10
   # Seconds in each sample interval:          10
   # Minutes between sample intervals:         30
   Frame sampling rate:                        1 out of 10

    -Broadcast Manager Configuration-
   IP BCM:                                     Disabled
   IPX BCM:                                    Disabled
   NetBIOS BCM:                                Disabled
   BCM IP Cache Aging Time:                    5
   BCM IPX Cache Aging Time:                   3
   BCM NetBIOS Cache Aging Time:               15
   Token Ring Source Route Management:         Disabled
   No BCM Static Entries defined
```

Note:

1 Redundancy is enabled on 8210-B

2 8210-B is configured in Backup LES/BUS mode

Figure 332. Enabling LES/BUS Redundancy on 8210-B

9 Restart 8210-A and verify the configuration

Restart 8210-A to activate the new configuration

```
Config>
*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 333 (Part 1 of 3). Restart and Verify 8210-A

Verify that the primary LES is active

```
*
*talk 5
+
+network 0
ATM Console
ATM+le-services
LE-Services Console
LE-SERVICES+work TR_ELAN_8210_1
LE-Services Console for an existing LES-BUS Pair
EXISTING LES-BUS ′TR_ELAN_8210_1′+list
ELAN Name:                TR_ELAN_8210_1
   ELAN Type:             Token Ring
   ATM Device number:     0
   # of Proxy LEC′s:      0
   # of Non-Proxy LEC′s:  0
   LES ATM Address:
   3909851111111111111111010140008210000A02

-Status-
   LES-BUS State:                      OPERATIONAL    1
   Redundancy VCC State:               ESTABLISHED    2
   Major Reason LES-BUS was last Down:  unknown err code
   Minor Reason LES-BUS was last Down:  none
   LES-BUS State last changed at:       00.00.04.42   (System Up Time)
   LES-LEC Status Table changed at:     00.00.00.00   (System Up Time)
   BUS-LEC Status Table changed at:     00.00.00.00   (System Up Time)
   UNI Version:                         3.1
   IP BCM:                              INACTIVE
   IPX BCM:                             INACTIVE
   NetBIOS BCM:                         INACTIVE
   Token Ring Source Route Management:  INACTIVE
```

Figure 333 (Part 2 of 3). Restart and Verify 8210-A

```
-Current Configuration-
   LES-BUS Enabled/Disabled:                   Enabled
   ATM Device number:                          0
   End System Identifier (ESI):                40.00.82.10.00.0A
   Selector Byte:                              0x02
   ELAN Type:       (S2)                       Token Ring
   Max Frame Size:  (S3)                       4544
   Control Timeout: (S4)                       120
   Max Frame Age:   (S5)                       1
   Validate Best Effort Peak Cell Rate (PCR):  No
   Control Distribute VCC Traffic Type:        Best Effort VCC
   Control Distribute VCC PCR in Kbps:         155000
   Control Direct VCC Max Reserved Bandwidth:  0
   Multicast Forward VCC Traffic Type:         Best Effort VCC
   Multicast Forward VCC PCR in Kbps:          155000
   Multicast Send VCC MAX Reserved Bandwidth:  0

    -LES-BUS Options-
   Security (LECS Validation of Joins):        Disabled
   Partition LE_ARP_REQUEST Forwarding Domain: Yes
   LE_ARP RESPONSE Destination:                One client
   Partition Unicast Frame Domain:             Yes
   Redundancy:                                 Enabled
   Redundancy Role:                            Primary LES-BUS
   ATM address of Backup LES-BUS:    3909851111111111111111010140008210000B02
   ATM address trace filter value:   0000000000000000000000000000000000000000
   ATM address trace filter mask:    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    -BUS Monitor Configuration-
   Monitor Host Usage of BUS:                  Disabled
   # Top Hosts to Record:                      10
   # Seconds in each sample interval:          10
   # Minutes between sample intervals:         30
   Frame sampling rate:                        1 out of 10

    -Broadcast Manager Configuration-
   IP BCM:                                     Disabled
   IPX BCM:                                    Disabled
   NetBIOS BCM:                                Disabled
   BCM IP Cache Aging Time:                    5
   BCM IPX Cache Aging Time:                   3
   BCM NetBIOS Cache Aging Time:               15
   Token Ring Source Route Management:         Disabled
   No BCM Static Entries defined
EXISTING LES-BUS 'TR_ELAN_8210_1'+
```

Note:

1 8210-A LES/BUS (Primary) is Operational

2 Redundancy VCC is established indicating that the primary LES/BUS is currently active.

Figure 333 (Part 3 of 3). Restart and Verify 8210-A

10 Restart 8210-B and verify the configuration

Restart 8210-B to activate the new configuration

```
Config>
*reload
Are you sure you want to reload the gateway? (Yes or [No]): yes
The configuration has been changed, save it? ([Yes] or No): yes
```

Figure 334 (Part 1 of 3). Restart and Verify 8210-B

Verify that backup LES/BUS is inactive

```
*
*talk 5
+
+network 0
ATM Console
ATM+le-services
LE-Services Console
LE-SERVICES+work TR_ELAN_8210_1
LE-Services Console for an existing LES-BUS Pair
EXISTING LES-BUS 'TR_ELAN_8210_1'+list
ELAN Name:               TR_ELAN_8210_1
   ELAN Type:            Token Ring
   ATM Device number:    0
   # of Proxy LEC's:     0
   # of Non-Proxy LEC's: 0
   LES ATM Address:      3909851111111111111111010140008210000B02

-Status-
   LES-BUS State:                      OPERATIONAL    1
   Redundancy VCC State:               ESTABLISHED    2
   Major Reason LES-BUS was last Down: unknown err code
   Minor Reason LES-BUS was last Down: none
   LES-BUS State last changed at:      00.32.09.53   (System Up Time)
   LES-LEC Status Table changed at:    00.00.00.00   (System Up Time)
   BUS-LEC Status Table changed at:    00.00.00.00   (System Up Time)
   UNI Version:                        3.1
   IP BCM:                             INACTIVE
   IPX BCM:                            INACTIVE
   NetBIOS BCM:                        INACTIVE
   Token Ring Source Route Management: INACTIVE
```

Figure 334 (Part 2 of 3). Restart and Verify 8210-B

```
-Current Configuration-
   LES-BUS Enabled/Disabled:                    Enabled
   ATM Device number:                           0
   End System Identifier (ESI):                 40.00.82.10.00.0B
   Selector Byte:                               0x02
   ELAN Type:       (S2)                        Token Ring
   Max Frame Size:  (S3)                        4544
   Control Timeout: (S4)                        120
   Max Frame Age:   (S5)                        1
   Validate Best Effort Peak Cell Rate (PCR):  No
   Control Distribute VCC Traffic Type:        Best Effort VCC
   Control Distribute VCC PCR in Kbps:         155000
   Control Direct VCC Max Reserved Bandwidth:  0
   Multicast Forward VCC Traffic Type:         Best Effort VCC
   Multicast Forward VCC PCR in Kbps:          155000
   Multicast Send VCC MAX Reserved Bandwidth:  0

    -LES-BUS Options-
   Security (LECS Validation of Joins):        Disabled
   Partition LE_ARP_REQUEST Forwarding Domain: Yes
   LE_ARP RESPONSE Destination:                One client
   Partition Unicast Frame Domain:             Yes
   Redundancy:                                 Enabled
   Redundancy Role:                            Backup LES-BUS
   ATM address trace filter value: 0000000000000000000000000000000000000000
   ATM address trace filter mask:  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    -BUS Monitor Configuration-
   Monitor Host Usage of BUS:                  Disabled
   # Top Hosts to Record:                      10
   # Seconds in each sample interval:          10
   # Minutes between sample intervals:         30
   Frame sampling rate:                        1 out of 10

    -Broadcast Manager Configuration-
   IP BCM:                                     Disabled
   IPX BCM:                                    Disabled
   NetBIOS BCM:                                Disabled
   BCM IP Cache Aging Time:                    5
   BCM IPX Cache Aging Time:                   3
   BCM NetBIOS Cache Aging Time:               15
   Token Ring Source Route Management:         Disabled
   No BCM Static Entries defined
EXISTING LES-BUS ′TR_ELAN_8210_1′+
```

Note:

1 8210-B LES/BUS (Backup) is Operational

2 Redundancy VCC is established indicating that the backup LES/BUS is currently inactive.

Figure 334 (Part 3 of 3). Restart and Verify 8210-B

11 Disconnect 8210-A and verify LES/BUS backup on 8210_B

Verify backup LES/BUS is active

```
*
* talk 5
+
+network 0
ATM Console
ATM+le-services
LE-Services Console
LE-SERVICES+work TR_ELAN_8210_1
LE-Services Console for an existing LES-BUS Pair
EXISTING LES-BUS ′TR_ELAN_8210_1′+list
ELAN Name:                TR_ELAN_8210_1
   ELAN Type:             Token Ring
   ATM Device number:     0
   # of Proxy LEC′s:      0
   # of Non-Proxy LEC′s:  0
   LES ATM Address:       39098511111111111111110101400082100000B02

-Status-
   LES-BUS State:                        OPERATIONAL    1
   Redundancy VCC State:                 IDLE           2
   Major Reason LES-BUS was last Down:   unknown err code
   Minor Reason LES-BUS was last Down:   none
   LES-BUS State last changed at:        00.32.09.53   (System Up Time)
   LES-LEC Status Table changed at:      00.00.00.00   (System Up Time)
   BUS-LEC Status Table changed at:      00.00.00.00   (System Up Time)
   UNI Version:                          3.1
   IP BCM:                               INACTIVE
   IPX BCM:                              INACTIVE
   NetBIOS BCM:                          INACTIVE
   Token Ring Source Route Management:   INACTIVE

-Current Configuration-
   LES-BUS Enabled/Disabled:                    Enabled
   ATM Device number:                           0
   End System Identifier (ESI):                 40.00.82.10.00.0B
   Selector Byte:                               0x02
   ELAN Type:       (S2)                        Token Ring
   Max Frame Size:  (S3)                        4544
   Control Timeout: (S4)                        120
   Max Frame Age:   (S5)                        1
   Validate Best Effort Peak Cell Rate (PCR):   No
   Control Distribute VCC Traffic Type:         Best Effort VCC
   Control Distribute VCC PCR in Kbps:          155000
   Control Direct VCC Max Reserved Bandwidth:   0
   Multicast Forward VCC Traffic Type:          Best Effort VCC
   Multicast Forward VCC PCR in Kbps:           155000
   Multicast Send VCC MAX Reserved Bandwidth:   0
```

Figure 335 (Part 1 of 2). Verify Backup LES/BUS

```
    -LES-BUS Options-
   Security (LECS Validation of Joins):        Disabled
   Partition LE_ARP_REQUEST Forwarding Domain: Yes
   LE_ARP RESPONSE Destination:                One client
   Partition Unicast Frame Domain:             Yes
   Redundancy:                                 Enabled
   Redundancy Role:                            Backup LES-BUS
   ATM address trace filter value: 0000000000000000000000000000000000000000
   ATM address trace filter mask:  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    -BUS Monitor Configuration-
   Monitor Host Usage of BUS:                  Disabled
   # Top Hosts to Record:                      10
   # Seconds in each sample interval:          10
   # Minutes between sample intervals:         30
   Frame sampling rate:                        1 out of 10

    -Broadcast Manager Configuration-
   IP BCM:                                     Disabled
   IPX BCM:                                    Disabled
   NetBIOS BCM:                                Disabled
   BCM IP Cache Aging Time:                    5
   BCM IPX Cache Aging Time:                   3
   BCM NetBIOS Cache Aging Time:               15
   Token Ring Source Route Management:         Disabled
   No BCM Static Entries defined
EXISTING LES-BUS 'TR_ELAN_8210_1'+
```

Note:

1 8210-B LES/BUS is operational.

2 Redundancy VCC is now idle, indicating that the backup LES/BUS is now active.

Figure 335 (Part 2 of 2). Verify Backup LES/BUS

11.4.21 ARP Server Redundancy

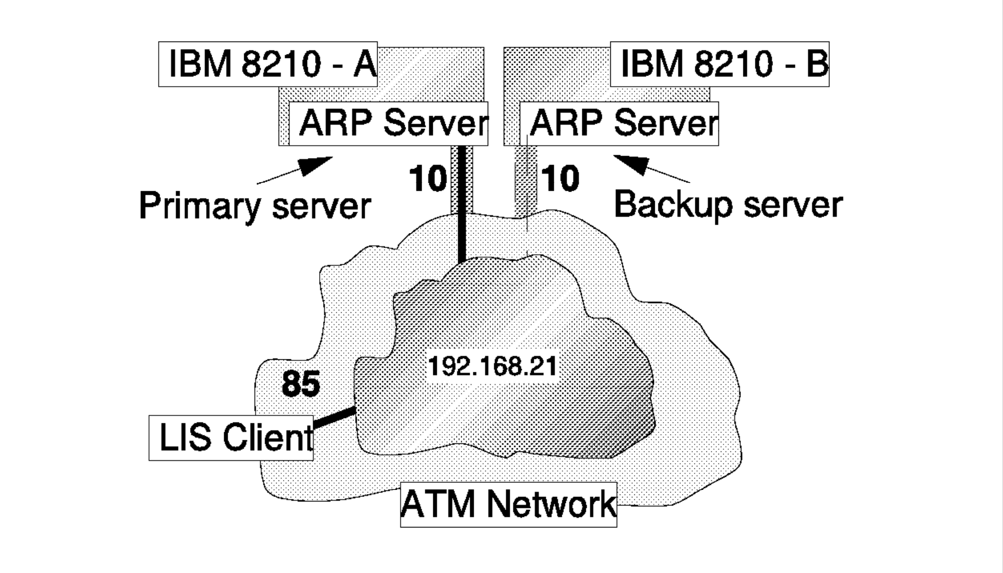


Figure 336. Multiple ARP Servers

Figure 336 depicts a fully automatic ARP server backup mechanism that is transparent to the LIS clients. It consists on two identically configured ARP servers having the same ATM address. Because the primary and backup ARP servers are using the same ATM address, the LIS clients can reconnect to the backup server when the primary server fails. Using the same ATM address requires that the ARP servers connect to the same ATM switch and use the same end system identifier (ESI) and selector byte.

Note: For a detailed discussion of this scenario, see 6.3.5, “Redundant ARP Server” on page 166.

This scenario works by virtue of ATM switches not allowing the same ESI to be registered twice, and the ESI registration retry procedure[3] for ARP servers (and clients).

The definitions required to realize this configuration are to define two 8210s with an exact equal configuration as described in 11.4.9, “IBM 8210 ARP Server” on page 336.

[3] The IBM 8210 Nways MSS Server retries ESI registration every 15 seconds, until successful.

11.4.22 IP Gateway Redundancy

This scenario combines the default IP gateway scenarios described in 7.7.1, “Default Gateway Redundancy - Classical IP” on page 201 and 7.7.2, “Default Gateway Redundancy - LAN Emulation” on page 202. It describes the configuration steps required where two 8210 LIS clients provide redundant default IP gateway support for other LIS clients, and two 8210 LE clients provide redundant default IP gateway support for other LE clients.

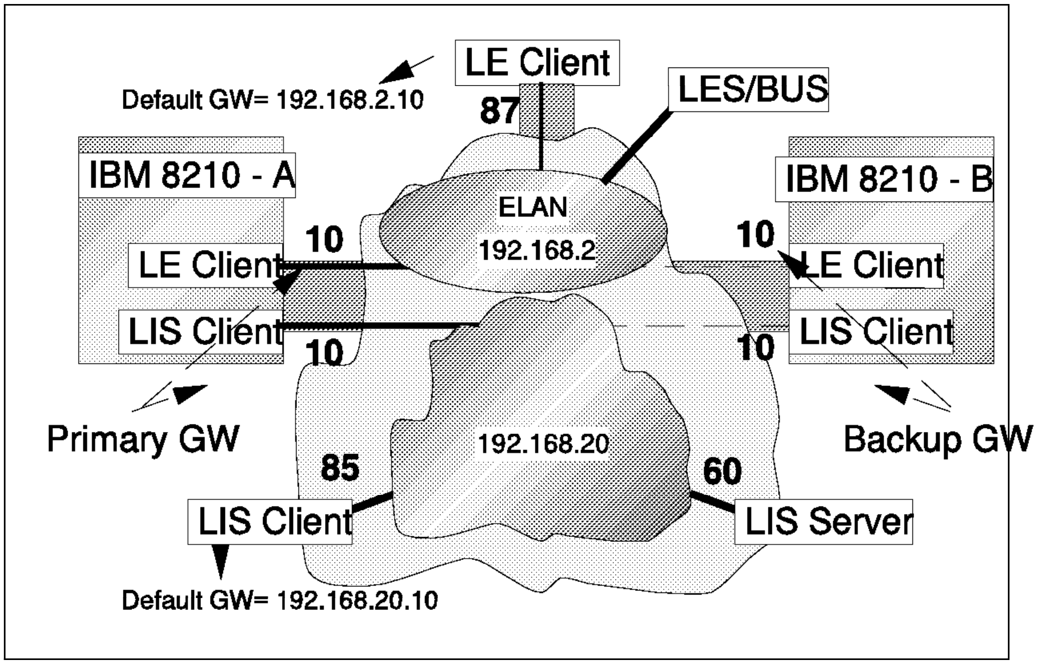


Figure 337. IP Gateway Redundancy

11.4.22.1 LIS Definitions

IP address 192.168.20.10 is associated with either the LIS client on 8210-A or the LIS client on 8210-B. The active 8210 LIS client provides IP routing functions between LIS 192.168.20 and ELAN 192.168.2.

Both 8210 LIS clients connect to the same ATM switch, and use the same end system identifier (ESI) and the same IP address (192.168.21.10). The required definitions are identical on both 8210s and have been discussed in 11.4.8, “IBM 8210 LIS Client” on page 332.

Note: To simplify the scenario an external ARP server is assumed. Both LIS clients could be defined as (redundant) ARP server as well.

11.4.22.2 ELAN Definitions

IP address 192.168.2.10 is associated with either an LE client active on 8210-A or an LE client on 8210-B. The active LE client provides IP routing functions between LIS 192.168.20 and ELAN 192.168.2.

Note: To simplify the scenario an external LES/BUS is assumed. Redundant LES/BUS and LECS functions could be defined as well on 8210-A and 8210-B.

Both 8210 LE clients connect to the same ATM switch, and use the same end system identifier (ESI) and the same IP address (192.168.2.10). The required definitions are identical on both 8210s.

Important

The active LE client and the active LIS client need to be on the same 8210. Therefore, make sure that LIS and LE clients all use the same ESI.

For a token-ring LE client use the definitions in 11.4.3, "IBM 8210 Token-Ring LE Client" on page 303. For an Ethernet LE client use the definitions in 11.4.4, "IBM 8210 Ethernet LE Client" on page 307. In addition, assign an IP address to the LE client on both 8210s:

```
*talk 6

Config>protocol ip
Internet protocol user configuration
IP config>add address 1                 1
New address [0.0.0.0]? 192.168.2.10
Address mask [255.255.255.0]?
IP config>exit
Config>
```

Note:

1 Assign the IP address to the proper logical interface.

Figure 338. Add IP Address

11.4.23 Spanning Tree Root Bridge Redundancy

This scenario describes the configuration steps required where two LE clients that have been configured for transparent bridging (TB) provide redundant spanning tree root bridge support.

Note: This configuration has been discussed in 5.10, “Redundant Spanning Tree Root Bridge” on page 118.

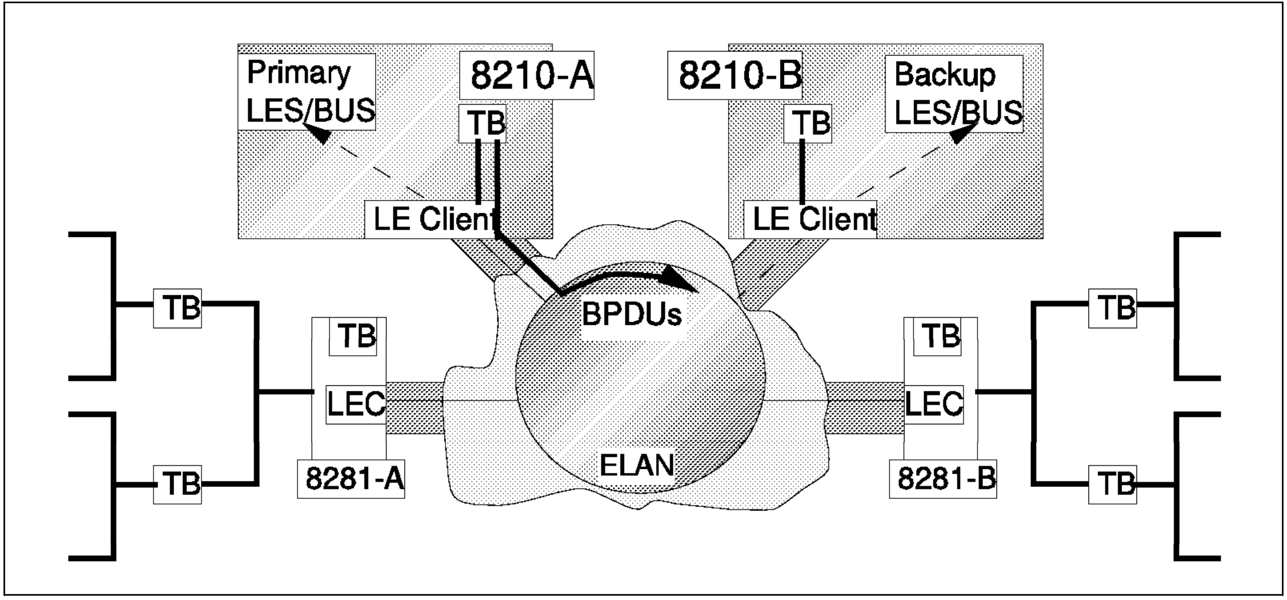


Figure 339. Spanning Tree Root Bridge Redundancy

This scenario describes the attachment of two 8210s to the same ATM switch. Using the Ethernet ELAN established by the 8210′s LES/BUS, legacy LAN interconnection is provided via multiple 8281s.

The LES/BUS and LE client on both 8210s have been identically configured, that is, with the same ESI and selector (SEL) value. The LE clients on both 8210s have been enabled for transparent bridging (TB) with an identical bridge configuration. By assigning the highest bridge priority we ensure that the 8210 becomes the spanning tree root bridge.

Important

The use of the same ESI for all LES/BUSs and LE clients, and connecting the 8210s to the same ATM switch ensures that, because only one 8210 can register the ESI, the active LES/BUS and LE client are always co-residents.

Note: Although a transparent bridge has been defined on the 8210, the 8281s communicate directly.

On the 8281s we use hard-coded LES addresses. As the second 8210 has the same ATM address as the first, the 8281s will re-register at the backup LES/BUS when the primary fails.

11.4.23.1 Important Timers

The 8210 LE client ESI registration retry timer can be up to 30 seconds, therefore the spanning tree root bridge backup might take up to (slightly more) than 30 seconds. It is important that an outage of this period is not interpreted by the 8281s as the root bridge being unavailable. Because of the 30 seconds backup delay, two timers need to be carefully set:

- The maximum aging (*max-age*) timer

  The max-age timer decides how long bridges will retain their image of the spanning tree before building up a new tree. This timer is reset each time when receiving BPDUs from the root bridge. The timeout period must be longer than the backup delay mentioned before. The max-age timer need only be considered on the root bridge, as all other bridges will accept the root bridge value and operate accordingly.

- The data direct VCC UP after control failure parameter

  The data direct VCC UP after control failure period configured on the 8281 bridges decides how long the 8281 will maintain its data direct VCC to other LE clients, after it has detected unavailability of the LES/BUS. The timeout period must be longer than the backup delay mentioned before, but to prevent looping data, it must be smaller than the max-age timer.

Based on the above arguments, we set the max-age timer on both 8210s (our root bridge) to 35 seconds, while we set the data direct VCC UP after control failure on all 8281s to 34 seconds.

Figure 340 on page 447 shows the important configuration parameters on the 8210s and the 8281s.

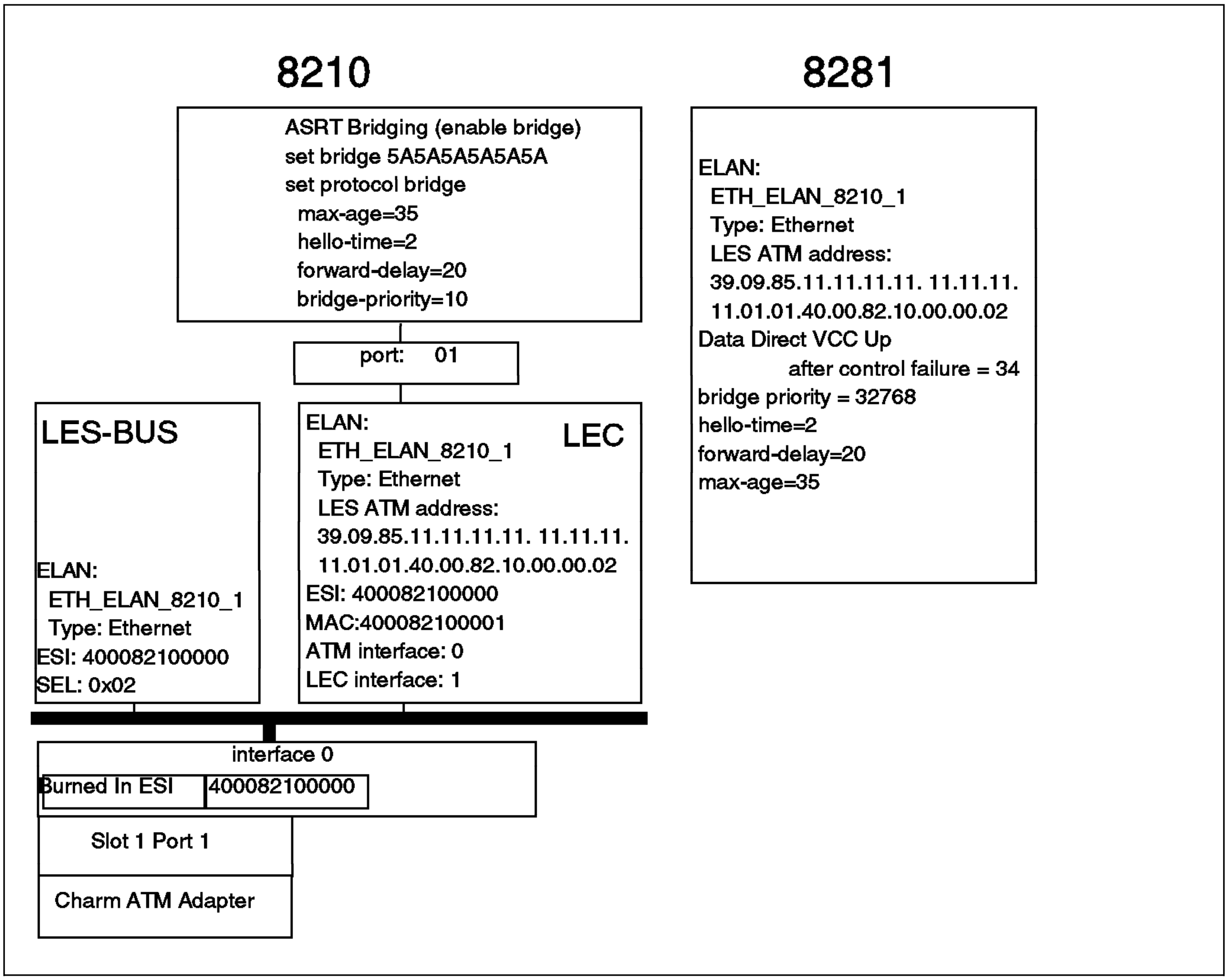


Figure 340. Redundant Root Bridge - Parameter Overview

To realize the scenario depicted in Figure 339 on page 445 we used the parameters shown in Figure 340. Critical parameters are:

- Max-age parameter on 8210s
- Data direct VCC UP after control failure on the 8281s
- Bridge priority on 8210s (value must be smaller than any other value coded in the network)

For the parameters not shown we used default values.

Note: We will show the configuration for a single 8210 as both 8210s are configured identical. The configuration steps required on the 8210s are:

1 **Configure the ATM interface**

```
* talk 6

Config>network 0
ATM user configuration
ATM Config>interface
ATM interface configuration
ATM Interface Config>set uni-version 3.1
ATM Interface Config>add esi
ESI in 00.00.00.00.00.00 form []? 40.00.82.10.00.00
ATM Interface Config>exit
ATM Config>
```

Figure 341. Configure ATM Interface

2 Configure the LES/BUS

```
*talk 6

Config>network 0
ATM user configuration
ATM Config>le-services
LAN Emulation Services user configuration
LE Services config>les-bus
ELAN Name (ELANxx) []? ETH_ELAN_8210_1
LES-BUS configuration
LES-BUS config for ELAN ′ETH_ELAN_8210_1′>add
Turn on Standard Event Logging for LES [yes]
Select ELAN type
        (1) Token Ring
        (2) Ethernet

Enter Selection:  [1]? 2
Select ESI
        (1) Use burned in ESI
        (2) 40.00.82.10.00.00

Enter Selection:  [1]? 2

Selector x00 is generally reserved for use by the LECS,
Selector x01 is generally reserved for use by the LECS Interface.

Enter selector (in hex) [2]? 2
Selection ″Add LES-BUS″ Complete
LES-BUS config for ELAN ′ETH_ELAN_8210_1′>exit
LE-Services config>exit
ATM Config>
```

Figure 342. Configure LES/BUS

3 Configure the LE client

```
ATM Config>le-client
ATM LAN Emulation Clients configuration
LE Client config>add ethernet
Added Emulated LAN as interface 1
LE Client config>config
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>set mac-address
Use adapter address for MAC? [Yes]: n
MAC address [00.00.00.00.00.00]? 40.00.82.10.00.01
Ethernet Forum Compliant LEC Config>set esi-address
Select ESI
   (1) Use burned in ESI
   (2) 40.00.82.10.00.00

Enter selection [1]? 2
Selector 0x2 is already in use on this interface
The selector has been changed to 0x4
Ethernet Forum Compliant LEC Config>exit
Config>
```

Set the LES/BUS ATM address for the LEC

```
Ethernet Forum Compliant LEC Config>set les
LES ATM address in 00.00.00.00.00.00:... form []?
39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.82.10.00.00.02
Ethernet Forum Compliant LEC Config>set auto no
Ethernet Forum Compliant LEC Config>exit
LE Client config>exit
ATM Config>exit
Config>
```

Figure 343. Configure LE Client

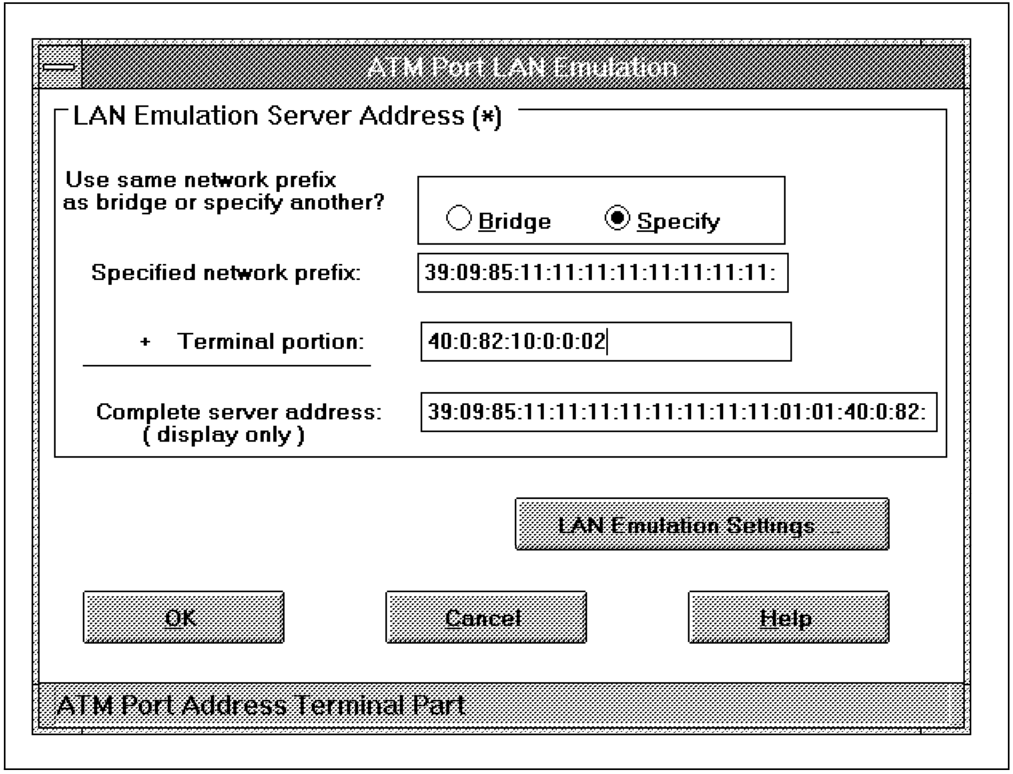
4 Configure the spanning tree root bridge

```
Config>protocol ASRT
Adaptive Source Routing Transparent Bridge user configuration
ASRT config>enable bridge
ASRT config>set bridge tb
Bridge Address (in 12-digit hex) []? 5A5A5A5A5A5A
ASRT config>set protocol bridge tb
Bridge-Max-Age [20]? 35
Bridge-Hello-Time [2]? 2
Bridge-Forward-Delay [15]? 20
Bridge-Priority [32767]? 10
ASRT config>exit
Config>
```

Figure 344. Configure Root Bridge

During configuration of the 8281s we made sure that:

1 Every 8281 uses a different ESI and has specified the proper LES address (see Figure 345 on page 450).



LAN Emulation Server Address (*)

Use same network prefix as bridge or specify another? ○ Bridge ● Specify

Specified network prefix: 39:09:85:11:11:11:11:11:11:11:11:

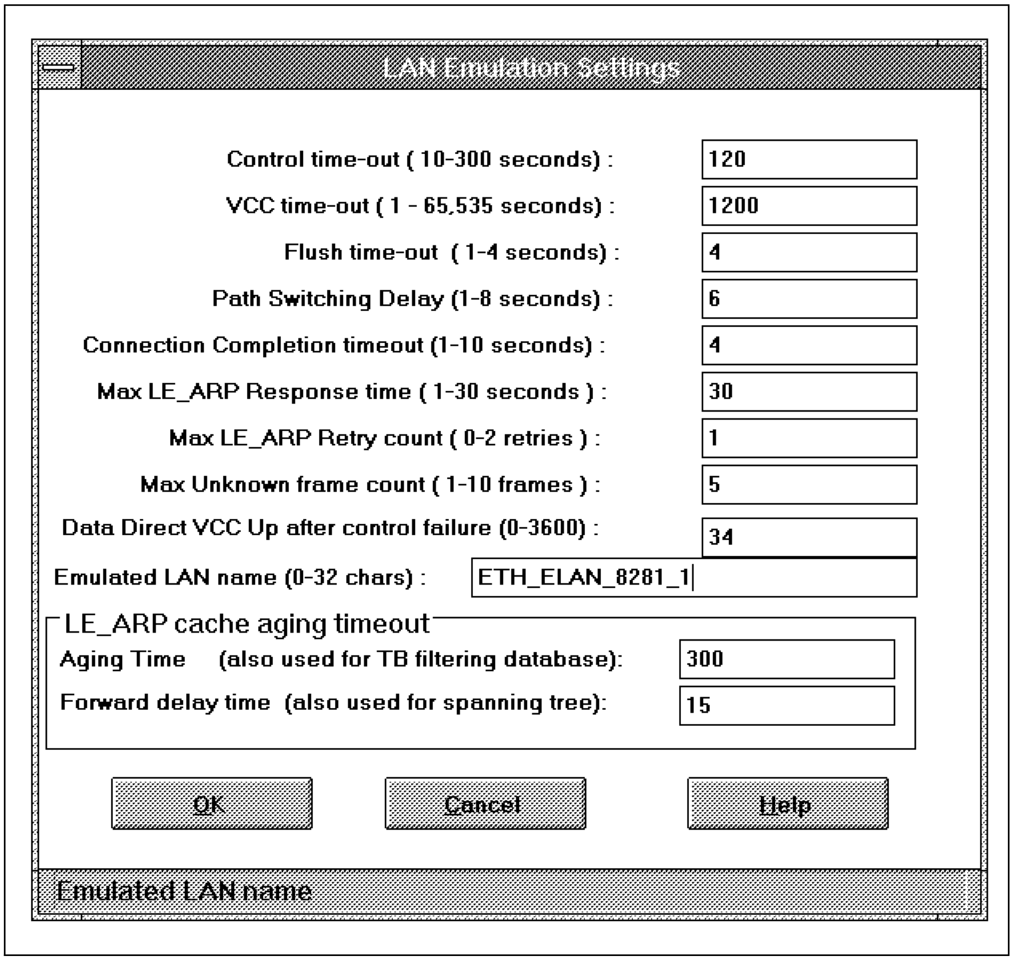
+ Terminal portion: 40:0:82:10:0:0:02

Complete server address: (display only) 39:09:85:11:11:11:11:11:11:11:11:01:01:40:0:82:

ATM Port Address Terminal Part

Figure 345. 8281 LES ATM Address

2 Every 8281 has a data direct VCC UP after failure of 30 seconds (see Figure 346 on page 451).



Control time-out (10-300 seconds) : 120

VCC time-out (1 - 65,535 seconds) : 1200

Flush time-out (1-4 seconds) : 4

Path Switching Delay (1-8 seconds) : 6

Connection Completion timeout (1-10 seconds) : 4

Max LE_ARP Response time (1-30 seconds) : 30

Max LE_ARP Retry count (0-2 retries) : 1

Max Unknown frame count (1-10 frames) : 5

Data Direct VCC Up after control failure (0-3600) : 34

Emulated LAN name (0-32 chars) : ETH_ELAN_8281_1

LE_ARP cache aging timeout

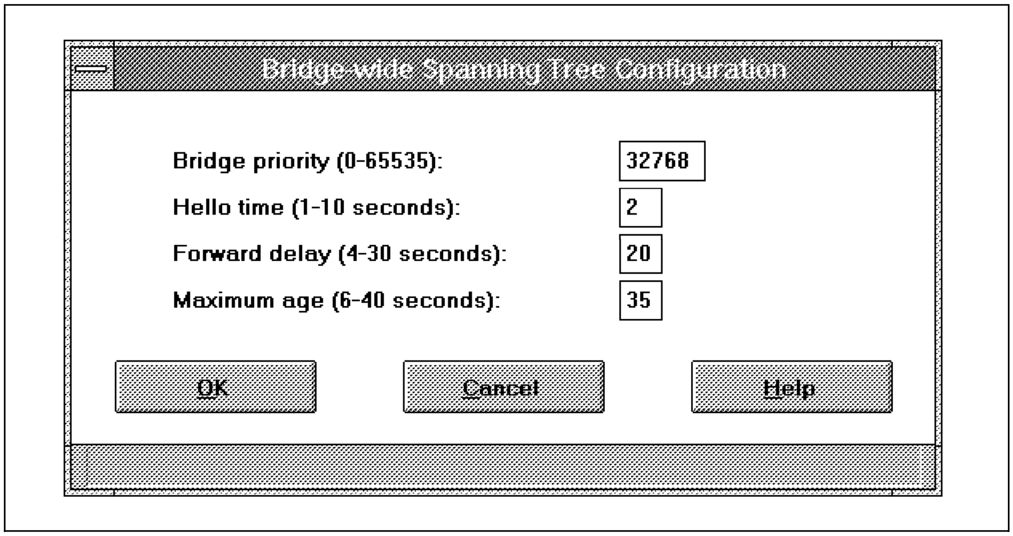
Aging Time (also used for TB filtering database): 300

Forward delay time (also used for spanning tree): 15

Figure 346. 8281 Data Direct VCC Timeout

Note: The emulated LAN name is irrelevant as the 8281s connect directly to the LES, and no security is enforced.

3 Every 8281 has a bridge priority lower than the 8210 (means a higher value) (see Figure 347).



Bridge priority (0-65535): 32768

Hello time (1-10 seconds): 2

Forward delay (4-30 seconds): 20

Maximum age (6-40 seconds): 35

Figure 347. 8281 Spanning Tree Parameters

Note: The max-age parameter is irrelevant as long as any of the 8210s operate as the root bridge.

11.5 Putting Things Together

The simple example scenarios in the previous sections can be used to build more complex environments.

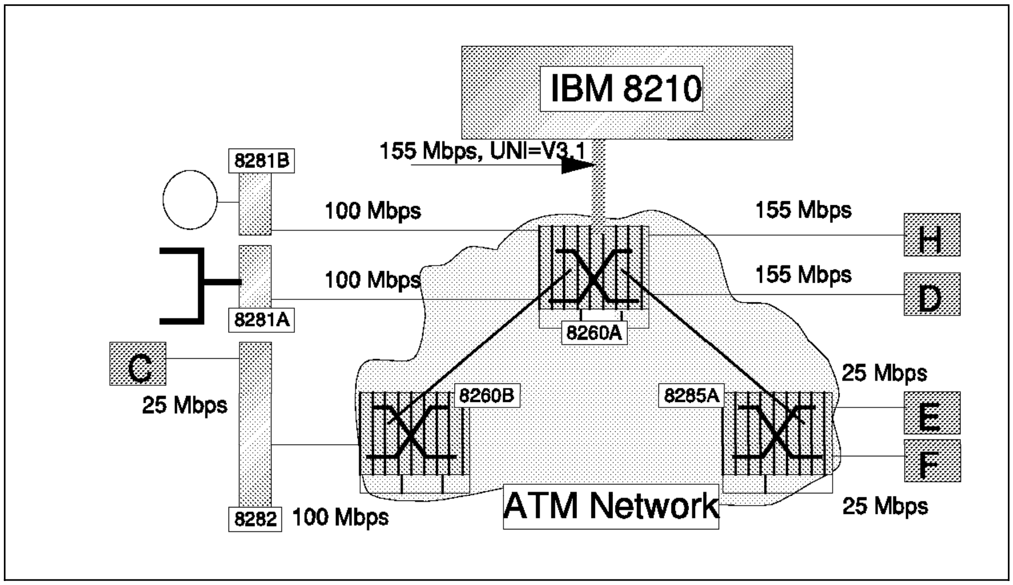


Figure 348. A More Complex Example

Figure 348 depicts an IBM 8210 Nways MSS Server that is connected to an ATM network consisting of three ATM switches: two 8260s and one 8285. Connected to the network we have a number of directly ATM-attached workstations (D, E, F, and H) and a workstation (C) connected via an IBM 8282 ATM Workgroup Concentrator. Furthermore, we are using two IBM 8281 ATM LAN Bridges to bridge an Ethernet and a token-ring legacy LAN to an Ethernet and token-ring ELAN, respectively.

The 8210 attaches to an ATM switch on which UNI V3.1 has been defined. On the 8210 we have specified UNI autodetection.

All, except one, workstations are running OS/2 and are equipped with an ATM adapter that supports Forum Compliant LAN emulation. Workstation H is an RS/6000 that performs ARP server functions.

Figure 349 on page 453 depicts a logical view of the network in which the following 8210 functions can be identified:

1 LAN emulation

a. LECS functions for six ELANs

1) Token-ring ELAN ‘TR_ELAN_8285’
2) Ethernet ELAN ‘ETH_ELAN_8285’
3) Token-ring ELAN ‘TR_ELAN_8210_1’
4) Token-ring ELAN ‘TR_ELAN_8210_2’
5) Ethernet ELAN ‘ETH_ELAN_8210_1’
6) Ethernet ELAN ‘ETH_ELAN_8210_2’

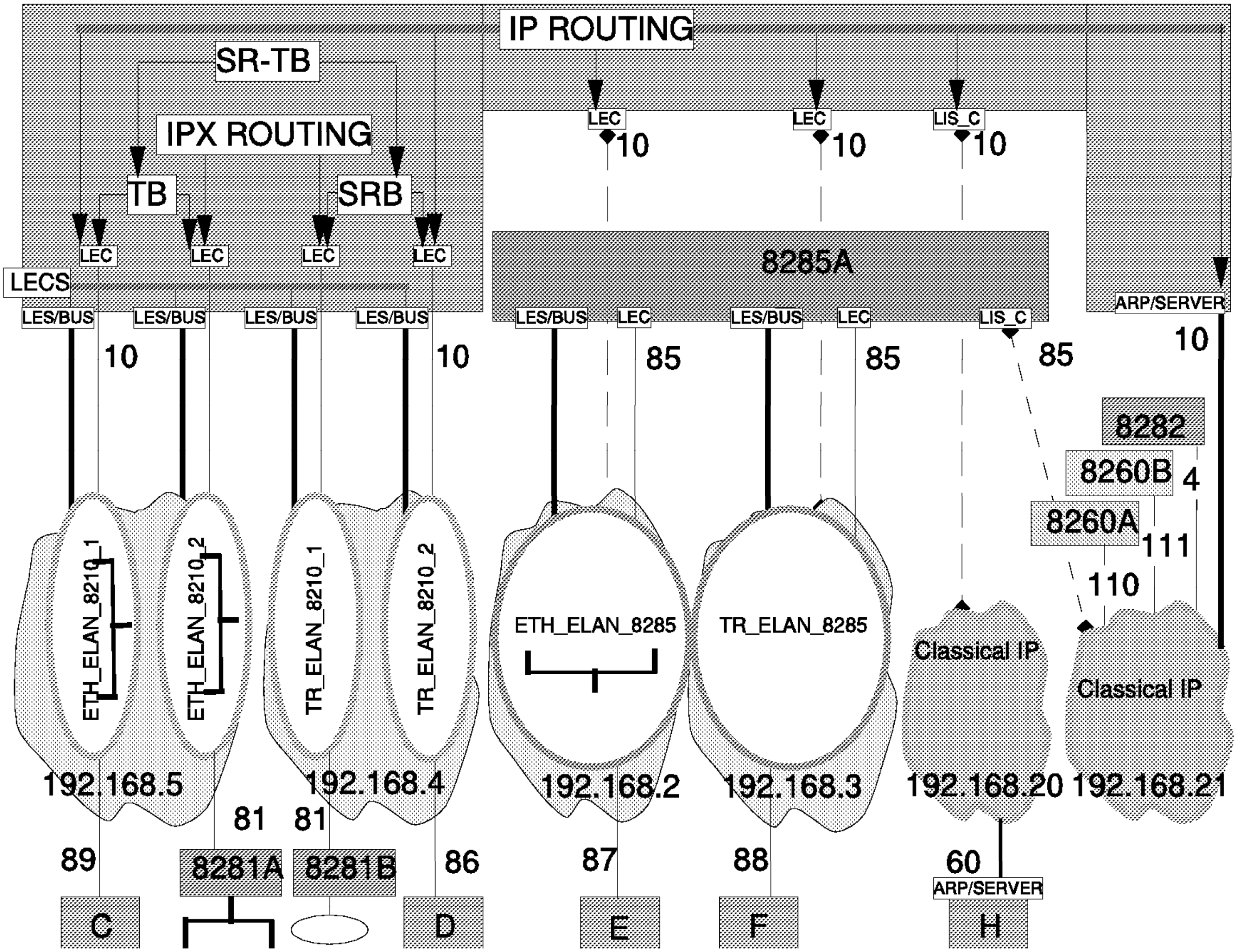


Figure 349. Client and Server Functions

The LECS assigns LE clients to ELANs based on ELAN name.

b. LES/BUS functions for four ELANs

1) Token-ring ELAN ‘TR_ELAN_8210_1’
2) Token-ring ELAN ‘TR_ELAN_8210_2’
3) Ethernet ELAN ‘ETH_ELAN_8210_1’
4) Ethernet ELAN ‘ETH_ELAN_8210_2’

The 8210 LES/BUS functions are used to establish each of these ELANs. BCM functions have been activated on all ELANs. SRM functions are used on the token-ring ELANs. LECS/LES security is not enabled.

c. LE client functions to six ELANs

1) Token-ring ELAN ‘TR_ELAN_8285’

The 8285 LES/BUS functions are used to establish this token-ring ELAN. To enable us to verify connectivity we have activated IP functions for the 8210 LE client, hereby the 8210 using IP address 192.168.3.10. The 8285 uses IP address 192.168.3.85, and workstation F, which is also a client on this ELAN, uses IP address 192.168.3.88.

2) Ethernet ELAN ‘ETH_ELAN_8285’

   The 8285 LES/BUS functions are used to establish this Ethernet ELAN. To enable us to verify connectivity we have activated IP functions for the 8210 LE client, hereby the 8210 is using IP address 192.168.2.10. The 8285 uses IP address 192.168.2.85, and workstation E, which is also a client on this ELAN, uses IP address 192.168.2.87.

3) Token-ring ELAN ‘TR_ELAN_8210_1’

   An 8210 LE client has been defined on this ELAN. Source-route bridging (SRB) has been enabled to token-ring ELAN ‘TR_ELAN_8210_2’ and IPX routing to ELAN ‘ETH_ELAN_8210_2’. 8281B uses IP address 192.168.4.81 on this ELAN.

4) Token-ring ELAN ‘TR_ELAN_8210_2’

   An 8210 LE client has been defined on this ELAN. To enable us to verify connectivity we have activated IP functions. Hereby the 8210 uses IP address 192.168.4.10. Workstation D, which is also a client on this ELAN, uses IP address 192.168.4.86. Source-route bridging (SRB) has been enabled to token-ring ELAN ‘TR_ELAN_8210_1’.

5) Ethernet ELAN ‘ETH_ELAN_8210_1’

   An 8210 LE client has been defined on this ELAN. To enable us to verify connectivity we have activated IP functions. Hereby the 8210 uses IP address 192.168.5.10. Workstation C, which is also a client on this ELAN, uses IP address 192.168.5.89. Transparent bridging (TB) has been enabled to Ethernet ELAN ‘ETH_ELAN_8210_2’.

6) Ethernet ELAN ‘ETH_ELAN_8210_2’

   An 8210 LE client has been defined on this ELAN. Transparent bridging (TB) has been enabled to Ethernet ELAN ‘ETH_ELAN_8210_1’ and IPX routing to token-ring ELAN ‘TR_ELAN_8210_1’. 8281A uses IP address 192.168.5.81 on this ELAN.

In addition to the previous functions, we have enabled translational (SR-TB) bridging between the SRB and TB bridges on the 8210.

2 Classical IP

a. LIS client for LIS 192.168.20

   Workstation (RS/6000) provides the ARP server function (IP address is 192.168.20.60) for this LIS, while the 8210 is connected as a LIS client (IP address 192.168.20.10).

b. ARP Server/LIS Client for LIS 192.168.21

   The 8210 provides the ARP server function (IP address is 192.168.21.10) for this LIS, while the 8282 (IP address 192.168.21.4), 8260-A (IP address 192.168.21.110), 8260-B (IP address 192.168.21.111), and 8285 (IP address 192.168.21.85) connect as clients.

3 IP routing is enabled between any of the interfaces on which an IP addresses has been defined:

a. LIS 192.168.20
b. LIS 192.168.21

c. Token-ring ELAN ‘TR_ELAN_8285’
d. Ethernet ELAN ‘ETH_ELAN_8285’
e. Token-ring ELAN ‘TR_ELAN_8210_2’
f. Ethernet ELAN ‘ETH_ELAN_8210_1’

4 IPX routing between:

a. Token-ring ELAN ‘TR_ELAN_8210_1’ and Ethernet ELAN ‘ETH_ELAN_8210_2’

5 Bridging

a. SRB bridging between token-ring ELAN ‘TR_ELAN_8210_1’ and token-ring ELAN ‘TR_ELAN_8210_2’

   The 8210 provides an LE client to both ELANs and we have enabled SRB functions between them.

b. TB bridging between Ethernet ELAN ‘ETH_ELAN_8210_1’ and Ethernet ELAN ‘TR_ELAN_8210_2’

   The 8210 provides an LE client to both ELANs and we have enabled TB functions between them.

c. Translational (SR-TB) bridging between the Ethernet and token-ring bridge ports on the 8210.

   The 8210 provides LE clients to all ELANs and we have enabled SR-TB functions between them.

6 SNMP

a. SNMP read/write access to all 8210 SNMP variables, using community name public, for all hosts within subnets 9.24.104 and 192.168. All SNMP traps are sent to 9.24.104.110.

11.5.1 Definitions Required

At first glance this scenario might appear complex. It should be emphasized, however, that although a considerable number of definition steps is required, each of these steps is pretty straightforward. To illustrate this we will not include all the definitions, but rather refer to the simple scenarios in the beginning of this chapter.

Important

Before starting to configure your 8210, translate your complex environment in terms of the components listed in 11.2, “Design Considerations” on page 289, identify the basic definition steps required for each, and configure accordingly.

In 11.5.2, “Resulting Definitions” on page 458 we have included a number of displays to show that the configuration actually works.

Definitions required are:

1 ATM Port

For details see 11.4.1, “IBM 8210 ATM Attachment with UNI Auto-Detection” on page 295.

Important

Make sure that the correct UNI version is defined. It is recommended to use user-defined ESIs.

2 LAN Emulation

a. LE client, and LECS functions for:

1) Token-ring ELAN ‘TR_ELAN_8285’

   For details see 11.4.5, “IBM 8210 Token-Ring LE Client with LECS” on page 310.

2) Ethernet ELAN ‘ETH_ELAN_8285’

   For details see 11.4.5, “IBM 8210 Token-Ring LE Client with LECS” on page 310. Instead of a token-ring LE client, define an Ethernet LE client. Make sure you define the proper ELAN name.

3) Token-ring ELAN ‘TR_ELAN_8210_1’

   For details see 11.4.6, “IBM 8210 Token-Ring LE Client, LES/BUS and LECS” on page 316.

4) Token-ring ELAN ‘TR_ELAN_8210_2’

   For details see 11.4.6, “IBM 8210 Token-Ring LE Client, LES/BUS and LECS” on page 316. Make sure you define the proper ELAN name.

5) Ethernet ELAN ‘ETH_ELAN_8210_1’

   For details see 11.4.7, “IBM 8210 Ethernet LE Client, LES/BUS, and LECS” on page 324.

6) Ethernet ELAN ‘ETH_ELAN_8210_2’

   For details see 11.4.7, “IBM 8210 Ethernet LE Client, LES/BUS, and LECS” on page 324. Make sure you define the proper ELAN name.

Important

Make sure that each LE client on the 8210:

- Uses a unique MAC address.
- Uses a unique selector (assuming all use the same ESI)

Note: Remember the logical interface numbers that are assigned by the configurator. They are needed during the definition of the higher layer (IP, IPX, bridging) functions.

b. LES/BUS functions for:

1) Token-ring ELAN ‘TR_ELAN_8210_1’

   For details see 11.4.6, “IBM 8210 Token-Ring LE Client, LES/BUS and LECS” on page 316.

2) Token-ring ELAN ‘TR_ELAN_8210_2’

   For details see 11.4.6, “IBM 8210 Token-Ring LE Client, LES/BUS and LECS” on page 316. Make sure you define the proper ELAN name.

3) Ethernet ELAN ‘ETH_ELAN_8210_1’

For details see 11.4.7, “IBM 8210 Ethernet LE Client, LES/BUS, and LECS” on page 324. No LECS/LES security definitions are required.

4) Ethernet ELAN ‘ETH_ELAN_8210_2’

For details see 11.4.7, “IBM 8210 Ethernet LE Client, LES/BUS, and LECS” on page 324. Make sure you define the proper ELAN name. No LECS/LES security definitions are required.

Important

Make sure that each LES/BUS instance on the 8210:

- Uses an user-defined ESI
- Uses an user-defined selector byte that is unlikely to change in future configurations

3 Classical IP

a. LIS client for one LIS

For details see 11.4.8, “IBM 8210 LIS Client” on page 332.

b. ARP Server/LIS Client for one LIS

For details see 11.4.9, “IBM 8210 ARP Server” on page 336.

Important

Make sure that the ARP server on the 8210:

- Uses an user-defined ESI
- Uses an user-defined selector byte that is unlikely to change in future configurations

4 IP routing

a. Token-ring ELAN ‘TR_ELAN_8210_2’ and Ethernet ELAN ‘ETH_ELAN_8210_1’

Define two LE clients, assign an IP address to each and IP routing is enabled. For details see 11.4.14, “IBM 8210 Ethernet to Token-Ring ELAN IP Routing” on page 362.

b. LIS 192.168.20 and LIS 192.168.21

Defining two LIS clients is sufficent to enable IP routing. For details see 11.4.12, “IBM 8210 LIS to LIS Routing” on page 351.

c. LIS 192.168.20 and token-ring ELAN ‘TR_ELAN_8210_2’

Defining the LIS and LE client, assign an IP address to the LE client, and IP routing is enabled. For details see 11.4.13, “IBM 8210 LIS to Token-Ring ELAN IP Routing” on page 356.

Important

When assigning IP addresses to LE client interface, make sure that the IP address is assigned to the proper logical interface.

5 IPX routing

a. IPX routing between token-ring ELAN ‘TR_ELAN_8210_1’ and Ethernet ELAN ‘ETH_ELAN_8210_2’.

For details see 11.4.15, “IBM 8210 Token-Ring to Ethernet ELAN IPX Routing” on page 373.

Important

When defining IPX routing, make sure that the IPX port definitions are applied to the proper logical interface.

6 Bridging

a. SRB bridging between token-ring ELAN ‘ETH_ELAN_8210_1’ and token-ring ELAN ‘TR_ELAN_8210_2’.

For details see 11.4.16, “IBM 8210 Source-Route Bridging (SRB)” on page 384.

b. TB bridging between Ethernet ELAN ‘ETH_ELAN_8210_1’ and Ethernet ELAN ‘TR_ELAN_8210_2’.

c. Translational (SR-TB) bridging between token-ring ELAN ‘TR_ELAN_8210_2’ and Ethernet ELAN ‘TR_ELAN_8210_1’

For details see 11.4.18, “IBM 8210 Source Route Translational Bridging (SR-TB)” on page 410.

Important

When defining bridging, make sure that the bridge port definitions are applied to the proper logical interface.

7 SNMP

For configuration details see 11.4.19, “SNMP Functions” on page 423.

11.5.2 Resulting Definitions

The definitions that result are:

1 ATM Port

```
ATM Interface Config>list config

                         ATM Configuration


  Interface (net) number =    0
  Maximum VCC data rate Mbps   =     155
  Maximum frame size     = 9234
  Maximum number of callers =  209
  Maximum number of calls = 1024
  Maximum number of parties to a multipoint call =  512
  Maximum number of Selectors that can be configured  =  200
  UNI Version = AUTO
  Packet trace = OFF
ATM Interface Config>list esi

          ESI         Enabled
  -----------------   -------
  40.00.82.10.00.00       YES
  50.00.82.10.00.00       YES
ATM Interface Config>
```

Note: ESI=X′400082100000′ is used for the LAN emulation functions, ESI=X′500082100000′ for Classical IP.

2 LAN Emulation

a. LECS

```
LECS config>list
LECS Detailed Configuration
   Lecs is                                      Enabled
   ATM Device number:                           0
   ESI:                                         40.00.82.10.00.00
   Selector:                                    0x00
   Validate Best Effort PCR:                    No
   Configuration Direct Max Reserved BW (Kbps): 0
   Maximum number of simultaneous VCCs:         128
   Idle VCC Timeout (in seconds):               60
   Trace ATM address value: 00.00.00.00.00.00.00.00.00.00.00.00.00.00.
                            00.00.00.00.00.00
   Trace ATM address mask:  FF.FF.FF.FF.FF.FF.FF.FF.FF.FF.FF.FF.FF.FF.
                            FF.FF.FF.FF.FF.FF
```

```
LECS config>elans
Configuration of ELANs for LECS
LECS ELANs config>list

ELAN Listing...

                             Name  Type  Packet Size  Enabled
=================================  ====  ===========  =======
                     TR_ELAN_8285  TokR         4544      Yes
                    ETH_ELAN_8285  Ethe         1516      Yes
                   TR_ELAN_8210_1  TokR         4544      Yes
                   TR_ELAN_8210_2  TokR         4544      Yes
                  ETH_ELAN_8210_1  Ethe         1516      Yes
                  ETH_ELAN_8210_2  Ethe         1516      Yes
LECS ELANs config>exit
```

```
LECS config>policies
LECS POLICIES configuration
LECS POLICIES config>list

Policy Listing...

Enabled  Priority  Type
=======  ========  ==========
    Yes        10  byElanNm
LECS POLICIES config>
```

b. LES/BUSs

```
LE Services config>list
List of Configured LES-BUS(s)

ELAN Type  (E=Ethernet/802.3, T=Token Ring/802.5)
| Interface #
| | Enabled                                              Max.
| | |                                                    Frame   Redundancy
| | | ELAN Name                       LES ESI       Sel  Size    Role
- - - ------------------------------- ------------  ---  -----   ----------
T 0 Y TR_ELAN_8210_1                  400082100000  xA1   4544   (disabled)
T 0 Y TR_ELAN_8210_2                  400082100000  xA2   4544   (disabled)
E 0 Y ETH_ELAN_8210_1                 400082100000  xB1   1516   (disabled)
E 0 Y ETH_ELAN_8210_2                 400082100000  xB2   1516   (disabled)
```

c. LE clients

```
LE Client config>list

   ATM Forum Compliant Emulated LANs
------------------------------------------------------
  ATM interface number = 0
  LEC interface number = 1
  Emulated LAN type    = Token Ring Forum Compliant
  Emulated LAN name    = TR_ELAN_8285
------------------------------------------------------
  ATM interface number = 0
  LEC interface number = 2
  Emulated LAN type    = Ethernet Forum Compliant
  Emulated LAN name    = ETH_ELAN_8285
------------------------------------------------------
  ATM interface number = 0
  LEC interface number = 3
  Emulated LAN type    = Token Ring Forum Compliant
  Emulated LAN name    = TR_ELAN_8210_2
------------------------------------------------------
  ATM interface number = 0
  LEC interface number = 4
  Emulated LAN type    = Token Ring Forum Compliant
  Emulated LAN name    = TR_ELAN_8210_1
------------------------------------------------------
  ATM interface number = 0
  LEC interface number = 5
  Emulated LAN type    = Ethernet Forum Compliant
  Emulated LAN name    = ETH_ELAN_8210_1
------------------------------------------------------
  ATM interface number = 0
  LEC interface number = 6
  Emulated LAN type    = Ethernet Forum Compliant
  Emulated LAN name    = ETH_ELAN_8210_2
```

3 Classical IP

```
ARP config>list atm-arp-client

ATM Arp Clients:
--------------------------------------------------
If: 0  Prot: 0  Addr: 192.168.20.10    ESI: 50.00.82.10.00.00  Sel: auto
Server: no   Refresh T/O: 5    AutoRefr: no    By InArp: yes  Validate PCR: no
Use Best Effort: yes/yes  (Control/Data)   Max B/W(kbps):      0
Cell Rate(kbps):  Peak:      0/     0    Sustained:      0/     0
Max SDU(bytes):   9188
--------------------------------------------------
If: 0  Prot: 0  Addr: 192.168.21.10    ESI: 50.00.82.10.00.00  Sel: 10
Server: yes  Refresh T/O: 20   AutoRefr: yes   By InArp: yes  Validate PCR: no
Use Best Effort: yes/yes  (Control/Data)   Max B/W(kbps):      0
Cell Rate(kbps):  Peak:      0/     0    Sustained:      0/     0
Max SDU(bytes):   9188
```

```
ARP config>list arp-server

ATM Arp Remote Server List:
   IP Address      Address / Sub Address
  192.168.20.10  39.09.85.11.11.11.11.11.11.11.11.01.01.40.00.00.60.00.01.00
```

4 IP definitions

```
IP config>list addresses
IP addresses for each interface:
   intf  0   192.168.20.10    255.255.255.0    Local wire broadcast, fill 1
             192.168.21.10    255.255.255.0    Local wire broadcast, fill 1
   intf  1   192.168.3.10     255.255.255.0    Local wire broadcast, fill 1
   intf  2   192.168.2.10     255.255.255.0    Local wire broadcast, fill 1
   intf  3   192.168.4.10     255.255.255.0    Local wire broadcast, fill 1
   intf  6   192.168.5.10     255.255.255.0    Local wire broadcast, fill 1
```

5 IPX definitions

```
IPX config>list

IPX globally                enabled
Host number (serial line)   000000000000
Router Name (IPXWAN)
NodeID (IPXWAN)             0
Maximum networks                  32
Maximum total route entries       32
Maximum routes per dest. network  1
Maximum services                  32
Maximum Network Cache entries     64
Maximum Local Cache entries       64

List of configured interfaces:
                Frame                     SAP nearest  Split
Ifc  IPX net #  Encapsulation             server reply Horizon      IPXWAN
  4         1   TOKEN-RING        MSB     Enabled      Enabled      N/A
  5         B   ETHERNET_802.3            Enabled      Enabled      N/A


RIP/SAP Timer Intervals
Ifc  IPX net #       SAP Interval(Minutes)   RIP Interval(Minutes)
  4         1                  1                      1
  5         B                  1                      1
IPX SAP Filter is: disabled
No IPX SAP Filter records in configuration.
IPX Access Controls are: disabled
No IPX Access Control records in configuration.
```

```
ASRT config>list bridge

                Source Routing Transparent Bridge Configuration
                ===============================================

Bridge:                  Enabled                 Bridge Behavior: SR<->TB
                   +----------------------------+
-------------------| SOURCE ROUTING INFORMATION |-------------------------------
                   +----------------------------+
Bridge Number:           01                      Segments:          2
Max ARE Hop Cnt:         14                      Max STE Hop cnt:   14
1: SRB:                 Active                  Internal Segment:  0xAAA
LF-bit interpret:        Extended
                   +-------------------+
-------------------| SR-TB INFORMATION |----------------------------------------
                   +-------------------+
SR-TB Conversion:        Enabled
TB-Virtual Segment:      0xFFF                   MTU of TB-Domain:  1470
                   +------------------------------------+
-------------------| SPANNING TREE PROTOCOL INFORMATION |-----------------------
                   +------------------------------------+
Bridge Address:          Default                 Bridge Priority:   32768/0x8000
STP Participation:       IEEE802.1d on TB ports, IBM-8209 and IBM-SRB proprietary
                                                 on SR ports
                   +-------------------------+
-------------------| TRANSLATION INFORMATION |----------------------------------
                   +-------------------------+
FA<=>GA Conversion:      Enabled                 UB-Encapsulation:  Disabled
                   +------------------+
-------------------| PORT INFORMATION |-----------------------------------------
                   +------------------+
Number of ports added: 6
Port:   1      Interface:      1      Behavior: No Bridging   STP:  Enabled
Port:   2      Interface:      2      Behavior: No Bridging   STP:  Enabled
Port:   3      Interface:      3      Behavior:    SRB Only   STP:  Enabled
Port:   4      Interface:      4      Behavior:    SRB Only   STP:  Enabled
Port:   5      Interface:      5      Behavior:    STB Only   STP:  Enabled
Port:   6      Interface:      6      Behavior:    STB Only   STP:  Enabled
```

```
ASRT config>list port
Port ID (dec)    : 128:01, (hex): 80-01
Port State       : Enabled
STP Participation: Enabled
Port Supports    : No Bridging
Assoc Interface  : 1
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:02, (hex): 80-02
Port State       : Enabled
STP Participation: Enabled
Port Supports    : No Bridging
Assoc Interface  : 2
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:03, (hex): 80-03
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Source-Route Bridging Only
SRB: Segment Number: 0x001       MTU:  4399     STE: Enabled
Assoc Interface  : 3
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:04, (hex): 80-04
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Source-Route Bridging Only
SRB: Segment Number: 0x002       MTU:  4399     STE: Enabled
Assoc Interface  : 4
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:05, (hex): 80-05
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only
Assoc Interface  : 5
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
Port ID (dec)    : 128:06, (hex): 80-06
Port State       : Enabled
STP Participation: Enabled
Port Supports    : Transparent Bridging Only
Assoc Interface  : 6
Path Cost        : 0
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

7 SNMP

```
SNMP Config>list all

SNMP is enabled
Trap UDP port: 162

          Community Name                  Access
-------------------------------  -------------------
public                            Read, Write, Trap



         Community Name            IP Address        IP Mask
-------------------------------  ---------------  ---------------
public                            192.168.0.0      255.255.0.0
                                  9.24.104.0       255.255.255.0
                                  9.24.104.110     255.255.255.255



         Community Name                   Enabled Traps
-------------------------------  --------------------------------
public                            Cold Restart, Warm Restart,
                                  Link Down, Link Up,
                                  Authentication Failure,
                                  EGP Neighbor Loss, Enterprise Specific



          Community Name                       View
-------------------------------  --------------------------------
public                            All


There are no views
```

Appendix A. ATM Forum-Compliant Frame Formats

The following sections provide you with the frame formats used in ATM Forum LAN emulation environment.

A.1 ATM Forum LAN Emulation Server Parameters

S1 **LAN Emulation Server′s ATM Address**

The LAN emulation server (LES) must know its own ATM addresses for the LAN emulation clients (LECs). The ATM address cannot be removed when any LEC is connected to the LES through it.

S2 **LAN Type**

This is the type of ATM emulated LAN. This can be either IEEE 802.3 (Ethernet) or IEEE 802.5 (token-ring).

S3 **Maximum Data Frame Size**

This is the maximum AAL-5 Service Data Unit (SDU) that the LAN emulation service can guarantee not to drop because it is too large. It is also the minimum AAL-5 SDU that every LEC must be able to receive. Valid values are 1516, 4544, 9234, or 18190 octets.

S4 **Control Timeout**

This parameter sets the period used for timing-out request/response control frame interactions. Once a LEC establishes a control direct VCC to the LES, the join phase must complete within the join time-out time. If this is not the case the LAN emulation service should release any control VCCs to that LEC thereby terminating the join phase.

S5 **Maximum Frame Age**

The broadcast and unknown server (BUS) must discard a frame if it has not sent the frame to all relevant multicast send VCCs or multicast forward VCCs within the maximum frame age following the BUS′s receipt of the frame over a multicast send VCC.
Values: minimum=1 second, maximum=4 seconds, default=1 second

S6 **Broadcast and unknown server′s ATM address**

A BUS must know at least one of its own ATM addresses for LECs to be able to establish connections to it. A BUS can have several ATM addresses. The address can be added while the BUS is operational but cannot be removed while any LEC has a connection to the BUS through the address.

A.2 ATM Forum LAN Emulation Client Parameters

C1 **LE Client′s ATM Address**

The primary ATM address used to connect to the LE server and the BUS. This must be known before the configuration and join phases can start and cannot change without restarting the configuration and join phases. The primary ATM address must be used to establish the LE client′s control direct VCC and multicast VCC, and must be specified as the SOURCE-ATM-ADDRESS in the client′s LE_JOIN_REQUESTs. An LE client may have additional ATM addresses for use with data direct VCCs. These address do not need to be known at join time and can be removed without restarting the join phase.

C2 LAN Type

The type of LAN that the client wishes to become or is a member of. This must be either Ethernet/IEEE 802.3, IEEE 802.5 or unspecified. This must not be unspecified after a successful join. This parameter must not be changed without terminating the client and returning to the Initial state.

C3 Maximum Data Frame Size

The maximum AAL-5 SDU size of a data frame that the LE client wishes to send on the multicast send VCC or to receive on the multicast forward VCC. This parameter also specifies the maximum AAL-5 SDU of all LE clients data direct VCCs. This value must not be unspecified after a successful join and cannot be changed without terminating the LE client and returning to the initial state.
Value: 1516, 4544, 9234, 18190 or unspecified.

C4 Proxy

This indicates whether the LE client may have remote unicast MAC addresses in C27. This must be known before the join phase can start and cannot be changed without restarting the configuration phase.

C5 ELAN Name

The identity of the emulated LAN the LE client wishes to join, or to which the LE client last joined. This may be unspecified before a join but can never be specified after a successful join.

C6 Local Unicast MAC Address(es)

Each LE client has zero or more local unicast MAC addresses. In an operational LE client every address in this variable must have been registered with the LE server. Two LE clients joined to the same emulated LAN cannot have the same local unicast MAC address. An LE client′s MAC address may change during normal operations.

C7 Control Timeout

Time-out period used for timing out most request/response control frame interactions, as specified elsewhere.
Value: minimum=10 seconds, maximum=300 seconds, default=120 seconds.

C8 Route Descriptors

Route descriptors exist only in source-routed IEEE 802.5 LE clients that are source-route bridges. All route descriptors in any given emulated LAN must be unique. An LE client may have zero or more route descriptors. Route descriptors can change during normal operation. If an LE client has route descriptors it must register all with the LE server.

C9 LE Server ATM Address

The ATM address of the LE server is used to establish the control direct VCC. This is obtained in the configuration phase from the LECS. It must be known before the join phase can start.

C10 Maximum Unknown Frame Count

Value: minimum=1, maximum=10 default=1.
(see parameter C11).

C11 Maximum Unknown Frame Time

Within the time period defined by the maximum unknown frame time an LE client will send no more than maximum unknown frame count frames to the BUS for a given unicast LAN destination, and it must also initiate the address resolution protocol to resolve that LAN destination.
Value: minimum=1second, maximum=60 seconds, default=1 second.

C12 VCC Time-Out Period

The LE client should release any data direct VCC that it has not used to transmit or receive any data frames for the length of the VCC timeout. This parameter only applies to SVC data direct VCCs.
Value: minimum=none, maximum=unlimited, default=20 minutes.

C13 Maximum Retry Count

An LE client must not retry an LE_ARP_REQUEST for a given LAN destination more than the maximum retry count times, after the first LE_ARP_REQUEST for the same frame′s LAN destination.
Value: minimum=0, maximum=2, default=1.

C14 LE Client Identifier

This is a unique identifier for the LE client assigned by the LE server. The LECID is placed in all control requests by the LE client and may be used for echo suppression on multicast data frames sent by that client. This value cannot change without first terminating the LE client and returning to the initial state.
Value: X′0001′ to X′FEFF′.

C15 LE Client Multicast MAC Address

Each LE client may have a list of multicast MAC addresses that it wishes to receive and pass upto the higher layers. The broadcast address should be included in the list.

C16 LE_ARP Cache

A table of entries, each of which establishes a relationship between a LAN destination external to the LE client and the ATM address to which data frames for that destination will be sent.

C17 Aging Time

The maximum time that an LE client will maintain an entry in its LE_ARP cache in the absence of verification of that relationship.
Value: minimum=10 seconds, maximum=300 seconds, default=300 seconds.

C18 Forward Delay Time

The maximum time that an LE client will maintain an entry for a non-local MAC address in its LE_ARP cache in the absence of a verification of that relationship, as long as the topology flag (C19) is true.
Value: minimum=4 seconds, maximum=30 seconds, default=15 seconds.

C19 **Topology Change**

Boolean indication that the LE client is using the forward delay time (C18), instead of the aging time (C17) to age non-local entries in its LE_ARP cache.

C20 **Expected LE_ARP Response Time**

The maximum time the LE client expects an LE_ARP_REQUEST/LE_ARP_RESPONSE cycle to take. It is used for retries and verifies.
Value: minimum=1 second, maximum=30 seconds, default=1 second.

C21 **Flush Timeout**

Time limit to wait to receive an LE_FLUSH_RESPONSE after the LE_FLUSH_REQUEST has been sent before taking recovery action.
Value: minimum=1 second, maximum=4 seconds, default=4 seconds.

C22 **Path Switching Delay**

The time since sending a frame to the BUS after which the LE client may assume that the frame has been either discarded or delivered to the recipient.
Value: minimum=1 second, maximum=6 seconds, default=8 seconds.

C23 **Local Segment ID**

The segment ID of the emulated LAN. Only required in IEEE 802.5 LE clients that are source-route bridges.

C24 **Multicast Send VCC Type**

Signalling parameter that should be used by the LE client when establishing the multicast send VCC. This is the method used by the LE client when specifying traffic parameters when it sets up the multicast send VCC for the emulated LAN.

C25 **Multicast Send VCC AvgRate**

Signalling parameter that should be used by the LE client when establishing the multicast send VCC. Forward and backward peak cell rate to be requested by the client when setting up the Multicast Send VCC if using variable bit encoding.

C26 **Multicast Send VCC PeakRate**

Signalling parameter that should be used by the LE client when establishing the multicast send VCC. Forward and backward peak cell rate to be requested when setting up the multicast send VCC when using either constant or variable bit rate coding.

C27 **Remote Unicast MAC Address(es)**

The MAC address for which this LE client will answer LE_ARP_REQUEST but that are not registered with the LE server. This list must be empty in any LE client that did not join the emulated LAN as proxy agent.

C28 **Connection Completion Timer**

Optional. In connection establishment this is the time period in which data or READY_IND message is expected from a calling party.
Value:Minimum=1 second, Maximum=10 seconds, Default=4 seconds.

A.3 Configuration Frame Format

Table 20. Configuration Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN Emulation protocol = X′01′.
3	1	VERSION	ATM LAN Emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0001′ LE_CONFIGURE_REQUEST X′0101′ LE_CONFIGURE_RESPONSE
6	2	STATUS	Always X′0000′ in requests. In responses refer to Table 27 on page 476 for a list of values.
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester and returned by the responder.
12	2	REQUESTER-LEC-ID	Always X′0000′ in requests, ignored on response.
14	2	FLAGS	Always X′0000′ when sent, ignored on receipt.
16	8	SOURCE-LAN-DESTINATION	MAC address or route descriptor of prospective LE client. May be encoded as ′not present′.
24	8	TARGET-LAN-DESTINATION	Always X′0000′ when sent, ignored on receipt.
32	20	SOURCE-ATM-ADDRESS	Primary ATM address of perspective LE client for which information is requested.
52	1	LAN-TYPE	X′00′ Unspecified X′01′ Ethernet/IEEE 802.3 X′02′ IEEE 802.5
53	1	MAXIMUM-FRAME-SIZE	X′00′ Unspecified X′01′ 1516 X′02′ 4544 X′03′ 9234 X′04′ 18190
54	1	NUMBER-TLVS	Number of Type/Length/Value element encoded in Request/Response.
55	1	ELAN-NAME-SIZE	Number of octets in ELAN_NAME - may be 0.
56	20	TARGET-ATM-ADDRESS	ATM address of the LE server to be used for the LE client described in the request if Configure Response and STATUS=′SUCCESS′, else X′00′.
76	32	ELAN-NAME	Name of emulated LAN.
108	4	ITEM_1-TYPE	Three octets of OUI, one octet identifier.
112	4	ITEM_1-LENGTH	Length in octets of VALUE field. Minimum=0
113	Variable	ITEM_1-VALUE	

A.4 Join Frame Format

Table 21. Join Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN Emulation protocol = X′01′.
3	1	VERSION	ATM LAN Emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0002′ LE_JOIN_REQUEST X′0102′ LE_JOIN_RESPONSE
6	2	STATUS	Always X′0000′ in requests. In responses refer to Table 27 on page 476 for a list of values.
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester and returned by the responder.
12	2	REQUESTER-LECID	Assigned LECID of joining client if join response and STATUS = ′SUCCESS′, else X′0000′.
14	2	FLAGS	Each bit of the FLAGS filed has a separate meaning if set: X′0080′ Proxy Flag: LE client server non-registered MAC addresses and therefore wishes to receive LE_ARP requests for non-registered LAN destinations.
16	8	SOURCE-LAN-DESTINATION	Optional MAC address to register as a pair with the SOURCE_ATM_ADDRESS.
24	8	TARGET-LAN-DESTINATION	Always X′00′ when sent, ignored on receipt.
32	20	SOURCE-ATM-ADDRESS	Primary ATM address of LE client issuing join request.
52	1	LAN-TYPE	X′00′ Unspecified X′01′ Ethernet/IEEE 802.3 X′02′ IEEE 802.5
53	1	MAXIMUM-FRAME-SIZE	X′00′ Unspecified X′01′ 1516 X′02′ 4544 X′03′ 9234 X′04′ 18190
54	1	NUMBER-TLVS	Always X′00′when sent, ignored on receipt.
55	1	ELAN-NAME-SIZE	Number of octets in ELAN_NAME. X′00′ indicates empty ELAN_NAME.
56	20	TARGET-ATM-ADDRESS	Always X′00′when sent, ignored on receipt.
76	32	ELAN-NAME	Name of emulated LAN.

A.5 Registration Frame Format

Table 22. Registration Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN Emulation protocol = X′01′.
3	1	VERSION	ATM LAN Emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0004′ LE_REGISTER_REQUEST X′0104′ LE_REGISTER_RESPONSE X′0005′ LE_UNREGISTER_REQUEST X′0105′ LE_UNREGISTER_RESPONSE
6	2	STATUS	Always X′0000′ in requests. In responses refer to Table 27 on page 476 for a list of values.
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester and returned by the responder.
12	2	REQUESTER-LECID	LECID of LE client issuing the register or unregister request and returned by the responder.
14	2	FLAGS	Always X′00′ when sent, ignored on receipt.
16	8	SOURCE-LAN-DESTINATION	Unicast MAC address or route descriptor LE Client is attempting to register.
24	8	TARGET-LAN-DESTINATION	Always X′00′ when sent, ignored on receipt.
32	20	SOURCE-ATM-ADDRESS	An ATM address of LE client issuing register or unregister request.
52	56	RESERVED	Always X′00′ when sent, ignored on receipt.

A.6 Address Resolution Frame Format

Table 23. LE_ARP Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN emulation protocol = X′01′.
3	1	VERSION	ATM LAN emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0006′ LE_ARP_REQUEST X′0106′ LE_ARP_RESPONSE
6	2	STATUS	Always X′0000′ in requests. In responses refer to Table 27 on page 476 for a list of values..
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester.
12	2	REQUESTER-LECID	LECID of LE client issuing the LE_ARP request.
14	2	FLAGS	Each bit of the FLAGS filed has a separate meaning if set: X′0001′ Remote address. The TARGET_LAN_DESTINATION is not register with the LE server.
16	8	SOURCE-LAN-DESTINATION	Source MAC address from data frame that triggered this LE_ARP sequence. May be encoded with ′not present′ LAN destination tag.
24	8	TARGET-LAN-DESTINATION	Destination unicast MAC address or next route descriptor for which an ATM address is being sought.
32	20	SOURCE-ATM-ADDRESS	ATM address of originator of LE_ARP request.
52	4	RESERVED	Always X′00′when sent, ignored on receipt.
56	20	TARGET-ATM-ADDRESS	X′00′ in LE_ARP request. ATM address of LE_Client responsible for target LAN destination in LE_ARP response.
76	32	RESERVED	Always X′00′when sent, ignored on receipt.

Table 24. LE_NARP Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN emulation protocol = X′01′.
3	1	VERSION	ATM LAN emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0008′ LE_NARP_REQUEST
6	2	STATUS	Always X′0000′.
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester.
12	2	REQUESTER-LECID	LECID of LE client issuing the LE_NARP request.
14	2	FLAGS	Always X′00′
16	8	SOURCE-LAN-DESTINATION	Not used. Encoded as X′00′.
24	8	TARGET-LAN-DESTINATION	Destination unicast MAC address or next route descriptor for which the target ATM address no longer applies.
32	20	SOURCE-ATM-ADDRESS	ATM address of originator of LE_NARP request.
52	4	RESERVED	Always X′00′when sent, ignored on receipt.
56	20	TARGET-ATM-ADDRESS	Target ATM address of LE_Client which was previously representing the target LAN destination.
76	32	RESERVED	Always X′00′when sent, ignored on receipt.

Table 25. Topology Change Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN emulation protocol = X′01′.
3	1	VERSION	ATM LAN emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0009′ LE_TOPOLOGY_REQUEST
6	2	STATUS	Always X′0000′.
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester.
12	2	REQUESTER-LECID	LECID of LE client issuing the topology change request.
14	2	FLAGS	Each bit of the FLAGS filed has a separate meaning if set: X′0100′ Topology Change Flag. A network topology change is in progress.
16	92	RESERVED	Always X′00′when sent, ignored on receipt.

A.7 Flush Frame Format

Table 26. LE_FLUSH Frame Format

Offset	Size	Name	Function
0	2	MARKER	Control Frame = X′FF00″.
2	1	PROTOCOL	ATM LAN emulation protocol = X′01′.
3	1	VERSION	ATM LAN emulation protocol version = X′01′.
4	2	OP-CODE	Type of request: X′0007′ LE_FLUSH_REQUEST X′0107′ LE_FLUSH_RESPONSE
6	2	STATUS	Always X′0000′ in requests. In responses refer to Table 27 on page 476 for a list of values.
8	4	TRANSACTION-ID	Arbitrary value supplied by the requester.
12	2	REQUESTER-LECID	LECID of LE client issuing the LE_ARP request.
14	2	FLAGS	Always 0 when sent, ignored on receipt.
16	8	SOURCE-LAN-DESTINATION	Always X′00′ when sent, ignored on receipt.
24	8	TARGET-LAN-DESTINATION	Always X′00′ when sent, ignored on receipt.
32	20	SOURCE-ATM-ADDRESS	ATM address of originator of the flush request.
52	4	RESERVED	Always X′00′when sent, ignored on receipt.
24	8	TARGET-LAN-DESTINATION	Destination unicast MAC address or next route descriptor for which an ATM address is being sought.
56	20	TARGET-ATM-ADDRESS	ATM address of LE_Client to which flush request is directed.
76	32	RESERVED	Always X′00′when sent, ignored on receipt.

A.8 Control Frame Status Values

Table 27. Control Frame Status Values

Code (dec)	Name	Meaning	Responses
0	Success	Successful response	All responses
1	Version not supported	Version field of request contains a value higher than that supported by the responder.	All responses
2	Invalid request parameters	The parameters given are incompatible with the ELAN	All responses
4	Duplicate LAN Destination registration.	SOURCE-LAN-DESTINATION duplicates a previously registered LAN destination.	Join or Register
5	Duplicate ATM address	SOURCE-ATM-ADDRESS duplicates a previously registered ATM address	Join or Register
6	Insufficient resources to grant request.	Responder is unable to grant request for reasons such a insufficient table space or ability to establish VCCs	Configure, Join or Register
7	Access denied	Request denied for security reasons.	Configure or Join
8	Invalid REQUESTER-ID	LECID field is not zero (Configure or Join) or is not LE Client′s LECID.	Configure, Join, Register, Unregister, ARP
9	Invalid LAN destination	LAN destination is a multicast address or on an Ethernet/IEEE 802.3 emulated LAN is a route descriptor.	Configure, Join, Register, ARP, Flush
10	Invalid ATM address	ATM address is not in a recognizable format.	Configure, Join, Register, ARP, Flush
20	No configuration	LE Client is not recognized.	Configure
21	LE_CONFIGURE Error	Parameters supplied give conflicting answers. May also be used to refuse service without giving a specific reason.	Configure
22	Insufficient Information.	LE Client has not provided sufficient information to allow the LECS to assign it to a specific emulated LAN	Configure

Appendix B. Dynamic IP Routing Protocols - Introduction

The IBM 8210 supports a number of IP routing (routing table maintenance) protocols. These allow the 8210 to exchange IP routing information with other 8210s and with other similarly capable IP routers. The following are supported:

- RIP Version 1
- OSPF Version 2
- BGP Version 4

RIP and OSPF are referred to as *interior gateway*[4] *protocols* (IGPs), while BGP is referred to as an *exterior gateway protocol* (EGP).

In the following sections the above routing protocols are discussed. For IBM 8210 IP configuration details, see Chapter 7, "MSS Server and IP Routing Protocols" on page 169.

B.1 Interior and Exterior Gateway Protocols

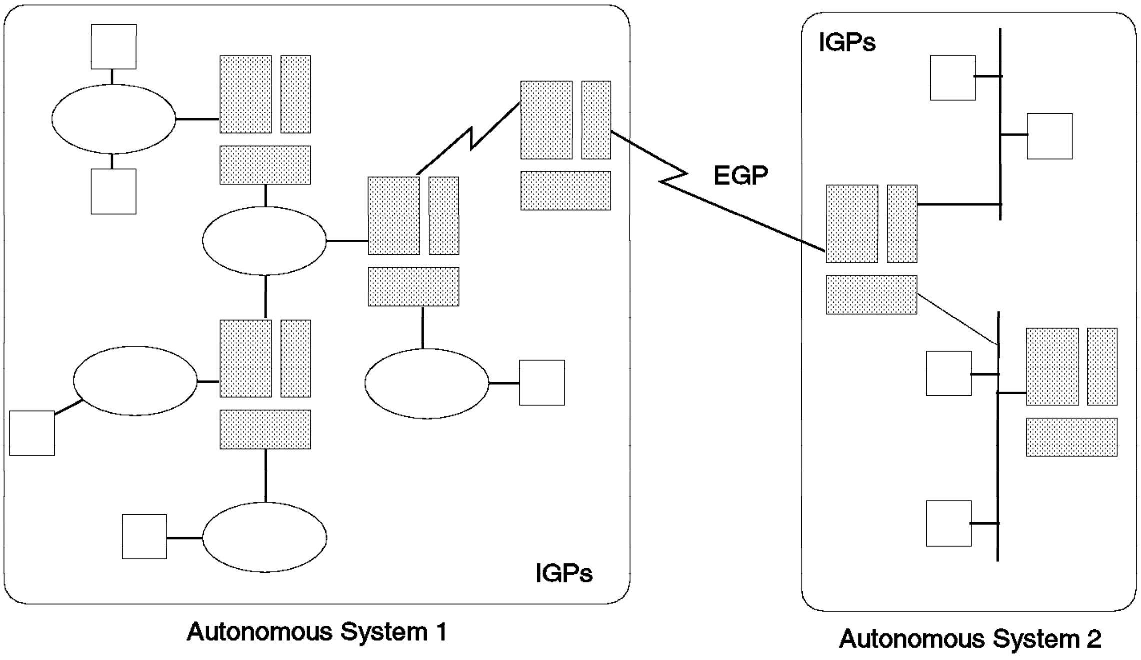


Figure 350. Autonomous Systems

Gateway protocols are referred to as interior or exterior depending on whether they are used within or between Autonomous Systems (ASs). Interior gateway protocols allow routers to exchange routing information within an AS. Exterior

[4] The term gateway is being used as is the convention for TCP/IP specialists (that is, to mean router). It is not being used in the strict ISO OSI sense.

gateway protocols allow the exchange of summary reachability information between separately administered ASs.

ASs are defined as logical portions of larger IP networks that are administered by single authorities. An AS would normally comprise the internetwork within an organization and would be designated as such to allow communication over public IP networks with ASs belonging to other organizations. It is mandatory to register an organization′s internetwork as an AS in order to use these public IP services.

Figure 350 on page 477 illustrates two ASs interconnected by routers. It shows how IGPs are used within the ASs and an EGP between them.

ASs must be registered publicly. If you require an AS number, or an IP network address, to allow your network to connect to public IP services you should contact:

Network Information Center
GSI
14200 Park Meadow Drive, Suite 200
Chantilly, VA 22021
USA
Tel: +1 703 802 4535

B.2 Choosing Gateway Protocols

Within an AS (or if you are building a private IP network) you are free to choose the interior gateway protocol, or combination of protocols, that best meets your needs.

Each interior gateway protocol, however, has different characteristics and selection must be carried out carefully to meet internetwork design requirements. The design considerations for selecting interior gateway protocols for an IP network using IBM 8210 IP routers are described in 7.2, “Using RIP” on page 175 and in 7.3, “Using OSPF” on page 180.

If you wish to communicate with other ASs you are once again, in principle, free to choose the exterior gateway protocol that best meets your needs. The interior gateway protocol used within an AS is not constrained by the choice of an exterior gateway protocol. However, there is synergy between some interior and exterior gateway protocols (for example, OSPF and BGP).

In practice AS-to-AS communication is governed by rules set by the administrators of the public Internet or by private IP service providers. An internetwork design must accommodate the exterior gateway protocols required by the IP service provider.

The design considerations for the exterior gateway protocol supported by the IBM 8210 IP router are described in 7.4, “Using BGP Version 4” on page 190.

B.3 Routing Algorithms

Interior and exterior gateway protocols currently implemented in the IBM 8210 use one of two generic classes of dynamic routing algorithm known as *distance vector* and *link-state* routing algorithms.

Dynamic routing algorithms allow routers to exchange route or link information, from which the best paths to reachable destinations in an internetwork are calculated.

Static routing may also be used to supplement dynamic routing.

B.3.1 Static Routing

Static routing requires that routes be configured manually into a router.

Normally manual configuration is to be avoided, particularly within an AS, but there are circumstances when static routing can be attractive:

- To define a default route, or a route that is not being advertised within a network
- To supplement or replace exterior gateway protocols:
  - When line tariffs between ASs are high or based on traffic volumes it may be desirable to avoid the cost of routing protocol traffic
  - If complex routing policies are to be implemented
  - To avoid disruption caused by faulty exterior gateways in other Autonomous Systems

B.3.2 Distance Vector Routing

Distance vector routing is still the most common type of routing encountered in IP internetworks.

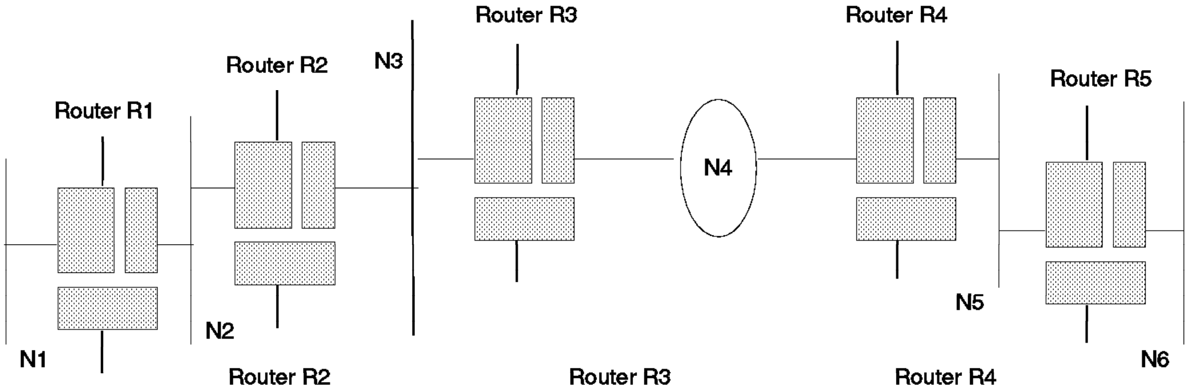
The principle behind distance vector routing is very simple. Each router in an internetwork maintains the distance from itself to every known destination in a *distance vector table*. Distance vector tables consist of a series of destinations (vectors) and costs (distances) to reach them and define the least costs to destinations at the time of transmission.

The distances in the tables are computed from information provided by neighbor routers that transmit their own distance vector tables across shared networks. The sequence of operations for doing this is as follows:

- Each router is configured with an identifier and the cost of each of its network links (the cost is normally fixed at 1 and hence equal to a single hop, but can reflect some other measurement taken for the link).
- Each router initializes with a distance vector table containing zero for itself, one for directly attached networks and infinity for every other destination.
- Each router periodically (typically every 30 seconds) transmits its distance vector table to each of its neighbors (it may also transmit it when a link first comes up or when the table changes).
- Each router saves the most recent table it receives from each neighbor and uses the information to calculate its own distance vector table.

- The total cost to each destination is calculated by adding the cost reported to it in a neighbor′s distance vector table to the cost of the link to that neighbor.
- The distance vector table (the routing table) for the router is then created by taking the lowest cost calculated for each destination.

Figure 351 shows the distance vector tables for three routers within a simple internetwork.



Router R2
Distance Vector
Table

Net	Next Hop	Metric
N1	R1	2
N2	=	1
N3	=	1
N4	R3	2
N5	R3	3
N6	R3	4

Router R3
Distance Vector
Table

Net	Next Hop	Metric
N1	R2	3
N2	R2	2
N3	=	1
N4	=	1
N5	R4	2
N6	R4	3

Router R4
Distance Vector
Table

Net	Next Hop	Metric
N1	R3	4
N2	R3	3
N3	R3	2
N4	=	1
N5	=	1
N6	R5	2

Figure 351. Distance Vector - Routing Table Calculation

The distance vector algorithm produces a stable routing table after a period directly related to the number of router hops across the network. This period is referred to as the *convergence time* for a network and represents the time it takes for distance vector information to traverse the network. It is possible in large internetworks for this time to be too long to be useful.

Routing tables are recalculated if a changed distance vector table is received from a neighbor, or if a link to a neighbor is found to have gone down. If a network link goes down, the distance vector tables that have been received over it are discarded and the routing table recalculated.

The chief advantage of a distance vector is that it is very easy to implement. There are also significant disadvantages; the main ones being the instability caused by old routes persisting in an internetwork, the long convergence time on large internetworks, the limit to the size of an internetwork imposed by maximum hop counts, and the fact that distance vector tables are always transmitted even if their contents have not changed.

Enhancements to the basic algorithm have evolved to overcome the first two of these problems. They are described in the following subsections.

B.3.2.1 Split Horizon

The basic distance vector algorithm will always allow a router to correctly calculate its distance vector table.

Using the example shown in Figure 352 you can see one of the problems of distance vector protocols known as *counting to infinity*.

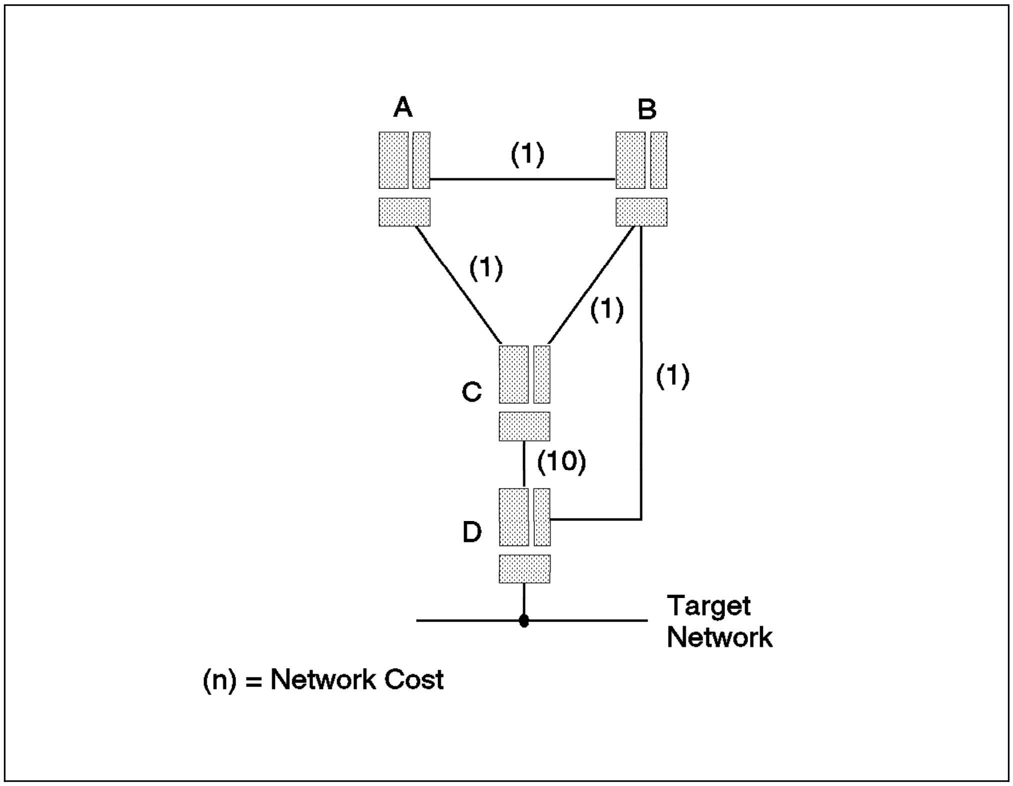


Figure 352. Counting to Infinity - Example Network

Counting to infinity occurs when a network becomes unreachable, but erroneous routes to that network persist because of the time for the distance vector tables to converge.

The example network shows four routers interconnected by five network links. The networks all have a cost of one except for that from C to D which has a cost of ten.

Each of the routers A, B, C and D have routes to all networks. If we show only the routes to the target network we will see they are as follows:

For D: Directly connected. Metric 1

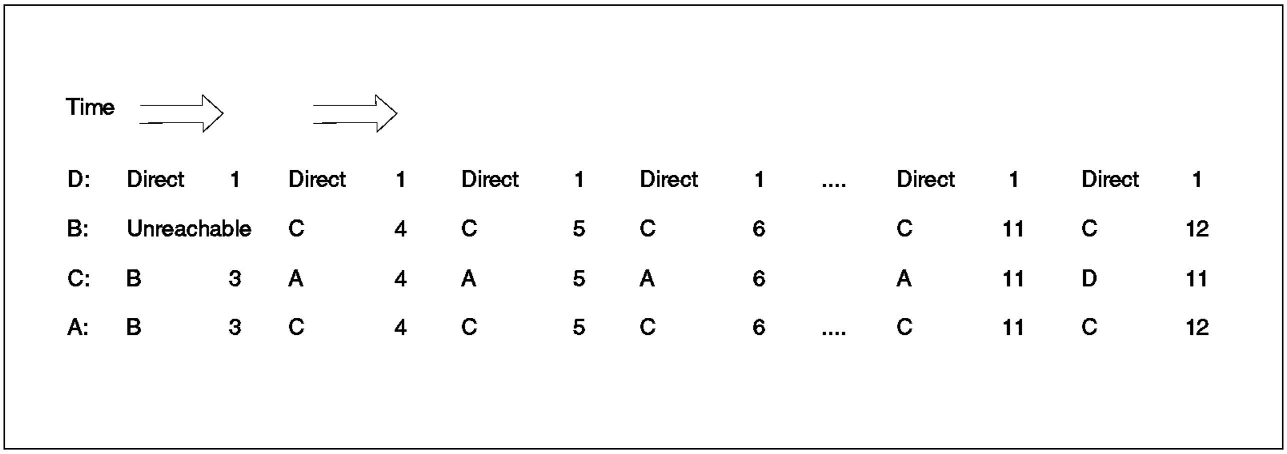
For B: Route via D. Metric 2

For C: Route via B. Metric 3

For A: Route via B. Metric 3

If the link from B to D fails, then all routes will adjust in time to use the link from C to D. The convergence time for this, however, may be considerable.

Distance vector tables begin to change when B notices that the route to D has become unavailable. Figure 353 on page 482 shows how the routes to the target network will change, assuming all routers send distance vector table updates at the same time.



Time

D:	Direct	1	Direct	1	Direct	1	Direct	1	....	Direct	1	Direct	1
B:	Unreachable		C	4	C	5	C	6		C	11	C	12
C:	B	3	A	4	A	5	A	6		A	11	D	11
A:	B	3	C	4	C	5	C	6	....	C	11	C	12

Figure 353. Counting to Infinity

The problem can be seen clearly. B is able to remove the failed route immediately because it times out the link. Other routers, however, have tables that contain references to this route for many update periods after the link has failed.

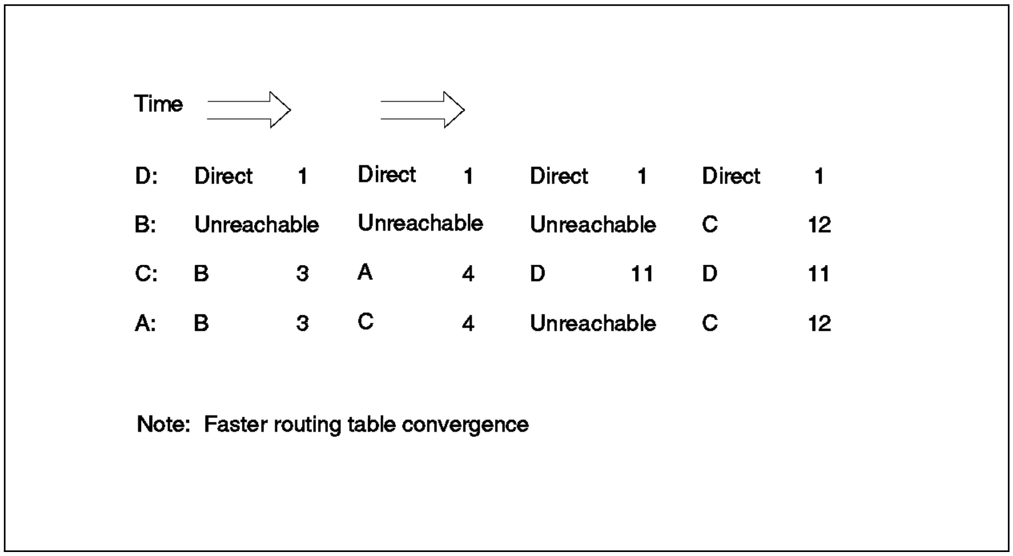
1. Initially A and C have a route to D via B.
2. Link from D to B fails.
3. A and C then send updates based on the route to D via B even after the link has failed.
4. B then believes it has a route to D via either A or C. It has not in reality as the routes are vestiges of the route via B which has failed.
5. A and C then see that the route via B has failed, but believe a route exists via one another.

Slowly the distance vector tables converge, but not until the metrics have counted up, in theory, to infinity. To avoid this happening practical implementations of distance vector have a low value for infinity; for example, RIP uses a maximum metric of 16.

The manner in which the metrics increment to infinity gives rise to the term *counting to infinity*. It occurs because A and C are engaged in an extended period of mutual deception, each claiming to be able to get to the target network D via one another.

Counting to infinity can easily be prevented if a route to a destination is never reported in the distance vector table sent to the neighbor from which the route was learned. *Split horizon* is the term used for this technique.

The incorporation of split horizon would modify the sequence of distance vector table changes to that shown in Figure 354 on page 483. The tables can be seen to converge considerably faster than in Figure 353.



Time ⇒ ⇒

D:	Direct	1	Direct	1	Direct	1	Direct	1
B:	Unreachable		Unreachable		Unreachable		C	12
C:	B	3	A	4	D	11	D	11
A:	B	3	C	4	Unreachable		C	12

Note: Faster routing table convergence

Figure 354. Split Horizon

B.3.2.2 Split Horizon with Poison Reverse

Poison reverse is an enhancement to split horizon whereby routes learned from a neighbor router are reported back to it but with a metric of infinity (that is, network unreachable).

The use of poison reverse is safer than split horizon alone because it breaks erroneous looping routes immediately.

If two routers receive routes pointing at each other, and they are advertised with a metric of infinity, the routes will be eliminated immediately as unreachable. If the routes are not advertised in this way they must be eliminated by the timeout that results from a route not being reported by a neighbor router for several periods (for example, six periods for RIP).

Poison reverse does have one disadvantage. It significantly increases the size of distance vector tables that must be exchanged between neighbor routers because all routes are included in the distance vector tables. While this is generally not a problem on LANs, it can cause real problems on point-to-point connections in large internetworks.

B.3.2.3 Triggered Updates

Split horizon with poison reverse will break routing loops involving two routers.

It is still possible, however, for there to be routing loops involving three or more routers. For example, A may believe it has a route through B, B through C and C through A. This loop can only be eliminated by the timeout that results from counting to infinity.

Triggered updates are designed to reduce the convergence time for routing tables and, hence, reduce the period during which such erroneous loops are present in a internetwork.

When a router changes the cost for a route in its distance vector table it must send the modified table immediately to neighbor routers. This simple mechanism ensures that topology changes in a network are propagated quickly, rather than at a rate dependent on normal periodic updates.

B.3.3 Link-State Routing

The growth in the size of internetworks in recent years has necessitated the replacement of distance vector routing algorithms with alternatives that address the shortcomings identified in Section B.3.2, “Distance Vector Routing” on page 479.

These new protocols have been based on *link-state* or shortest path first algorithms. The best example is the OSPF interior gateway protocol.

The principle behind link-state routing is straightforward, although implementation can be complex:

- Routers are responsible for contacting neighbors and learning their identities.
- Routers construct link-state packets that contain lists of network links and their associated costs.
- Link-state packets are transmitted to all routers in a network.
- All routers, therefore, have an identical list of links in a network and can construct identical topology maps.
- The maps are used to compute the best routes to all destinations.

Routers contact neighbors by sending *Hello* packets on their network interfaces. Hello packets are sent directly to neighbors on point-to-point links and non-broadcast networks. On LAN networks, Hello packets are sent to a predefined group or multicast IP address that can be received by all routers. Neighbors who receive Hellos from a router should reply with Hello packets that include the identity of that originating router.

Once neighbors have been contacted in this way, link-state information can be exchanged.

Link-state information is sent in the form of *link-state packets* (LSPs), also known as link-state advertisements. LSPs provide the database from which network topology maps can be calculated at each router. LSPs are normally sent only under the following specific circumstances:

- When a router discovers a new neighbor
- When a link to a neighbor goes down
- When the cost of a link changes

Once a router has generated an LSP, it is critical that it is received successfully by all other routers in a network. If this does not happen routers on the network will calculate network topology based on incorrect link-state information.

Distribution of LSPs would normally be on the basis of each router′s routing tables. However, this leads to a *chicken and egg* situation. Routing tables would rely on LSPs for their creation and LSPs would rely on routing tables for their distribution. A simple scheme called *flooding* overcomes this and ensures that LSPs are successfully distributed to all routers in a network.

Flooding requires that a router that receives an LSP must transmit it to all neighbors except the one from which it was received. All LSPs must be explicitly acknowledged to ensure successful delivery, and they are sequenced and time stamped to ensure duplicates are not received and retransmitted.

When a router receives an LSP it looks in its database to see the sequence number of the last LSP from the originator. If the sequence number is the same as, or earlier than, the sequence number of the LSP in its database, then the LSP is discarded. Otherwise the LSP is added to the database.

The flooding process ensures that all routers in a network have the same link-state information. All routers are then able to compute the same shortest path tree topology map for the network and, hence, select best routes to all destinations.

B.4 Routing Information Protocol (RIP)

The *routing information protocol* (RIP) is an interior gateway protocol defined in RFC 1058.

It is an IAB standard protocol; its status is elective. This means that it is one of several interior gateway protocols available, and it may or may not be implemented on a system. If a system does implement it, however, the implementation should be in line with the RFC.

RIP is based on the Xerox PUP and XNS routing protocols. The RFC was issued after many RIP implementations had been completed. For that reason some do not include all the enhancements to the basic distance vector routing protocol (such as poison reverse and triggered updates).

RIP is very widely used because the code (known as *ROUTED*) was incorporated on the Berkeley Software Distribution (BSD) UNIX operating system and in other UNIX systems based on it.

The next sections overview the RIP protocol. The IBM 8210 supports a full implementation of RIP Version 1. For details on the IBM 8210 IP implementation, see B.4.2, “IBM 8210 and RIP” on page 487.

B.4.1 Protocol Description

RIP is a standard distance vector routing protocol, as described in B.3.2, “Distance Vector Routing” on page 479.

RIP packets are transmitted onto a network in *User Datagram Protocol* (UDP) datagrams, which, in turn, are carried in IP datagrams. RIP sends and receives datagrams using UDP port 520. RIP datagrams have a maximum size of 512 bytes and tables larger than this must be sent in multiple UDP datagrams.

RIP datagrams are normally broadcast onto LANs using the LAN MAC All-Stations broadcast address and the IP network or subnetwork broadcast address. They are specifically addressed on point-to-point and multi-access non-broadcast networks, using the destination router IP address.

Routers normally run RIP in *active mode*; that is, advertising their own distance vector tables and updating them based on advertisements from neighbors. End nodes, if they run RIP, normally operate in *passive (or silent) mode*; that is, updating their distance vector tables on the basis of advertisements from neighbors, but not advertising them.

RIP specifies two packet types: *request* and *response*.

A request packet is sent by routers to ask neighbors to send part of their distance vector table (if the packet contains destinations), or all their table (if no destinations are specified).

A response packet is sent by routers to advertise their distance vector table in the following circumstances:

- Every 30 seconds
- In response to a request packet
- When distance vector tables change (if triggered updates are supported)

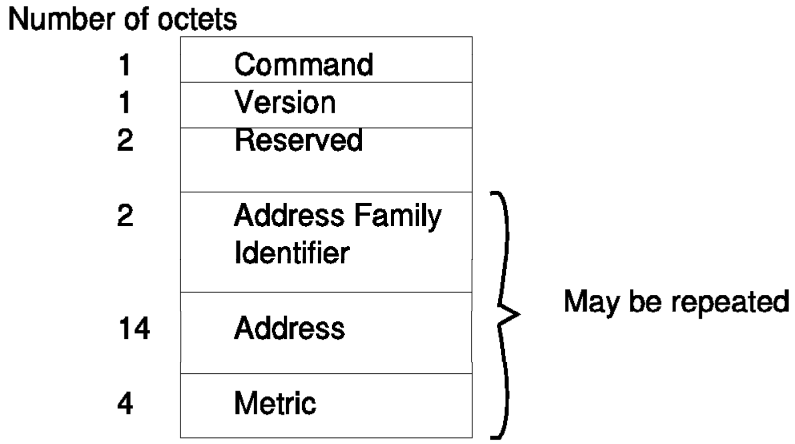


Figure 355. Generalized RIP Packet Format

Active and passive systems listen for all response packets and update their distance vector tables accordingly. A route to a destination, computed from a neighbor′s distance vector table, is kept until an alternate is found with lower cost, or it is not re-advertised in six consecutive RIP responses. In this case the route is timed out and deleted.

The RFC defines a packet format that can be used with different network protocols. It does this by specifying an *address family identifier* that defines the type (and hence the length) of the network address. The generalized format of RIP packets is shown in Figure 355.

RIP may be used, therefore, for protocols other than IP simply by setting the address family identifier. The RFC requires that RIP response handling discards entries for unsupported address families but processes entries for supported address families in the normal way.

When RIP is used with IP the address family identifier is 2, and the address fields are 4 bytes. To reduce problems of counting to infinity the maximum metric is 16 (unreachable) and directly connected networks are defined as having a metric of one.

The RIP packet format for IP is shown in Figure 356 on page 487.

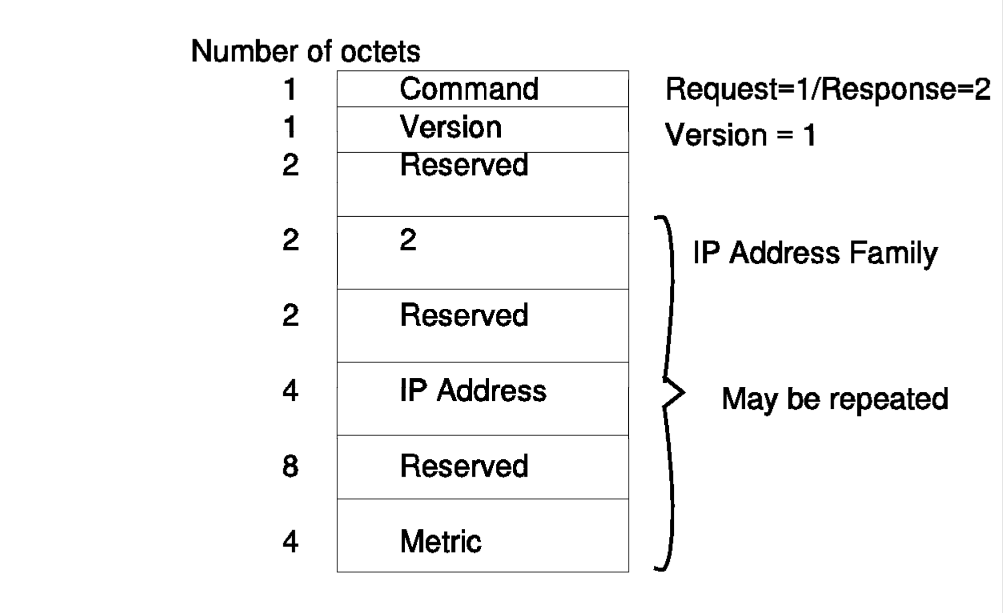


Figure 356. IP Specific RIP Packet Format

RIP makes no provision for passing subnet masks with its distance vector tables. A router receiving a RIP response must already have subnet mask information to allow it to interpret the network identifier and host identifier portions of the IP address correctly.

In the absence of subnet mask information a router will interpret routes as best it can. If it knows an IP network has a specific subnet mask it will interpret all other route information for that network on the basis of that single mask. If it receives a packet with bits set in the field that it regards as the host field it will interpret it as a route to a host with a mask of *255.255.255.255*.

The above makes it impossible for RIP to be used in an internetwork with variable length subnet masks.

B.4.2 IBM 8210 and RIP

The IBM 8210 supports RIP. The IBM 8210 implementation includes support for split horizon and poison reverse. They are logically exclusive as split horizon does not send routes back onto the subnet that they were received from while poison reverse advertises an infinite route back onto the net that the route was received from. The choice of neither split horizon nor poison reverse is not allowed. Poison reverse can be disabled per interface. To speed network convergence, triggered updates are sent.

B.4.2.1 Route Acceptance Policy

Per interface the user can configure whether or not:

- RIP will listen
- Network, subnet, and/or host routes are learned
- Default and/or static routes can be overridden with a learned route

Note: Default routes are indicated by a destination and mask of 0.0.0.0.

B.4.2.2 Route Advertisement Policy

Per interface the user can configure whether or not:

- RIP will advertised
- Network, subnet, and/or host routes are advertised
- Default and/or static routes are advertised

Subnets will be advertised if they are part of the same natural network (that is, class A, B, or C) as the interface′s IP address, and the route subnet mask must be the same as the interface′s subnet mask. Host routes (mask 255.255.255.255) are exempt from these rules.

For details on configuring 8210 IP RIP functions, see 7.2.1, "RIP Configuration on the IBM 8210" on page 176.

B.5 Open Shortest Path First (OSPF)

The *open shortest path first* (OSPF) V2 protocol is an interior gateway protocol defined in RFC 1583. A report on the use of OSPF V2 is contained in RFC 1246 - *Experience with the OSPF Protocol*.

It is an IAB standard protocol; its status is elective.

OSPF is important because it has a number of features not found in other interior gateway protocols. Support for these additional features makes OSPF the preferred choice for new IP internetwork implementations:

- Variable length subnet masks
- Alternate routes based on IP type of service (TOS)
- Equal cost multipath routes
- Faster network convergence in the presence of change, no counting to infinity
- Better scaling capability
- Use of local multicast to limit traffic to OSPF routers

The next sections overview the OSPF protocol. The IBM 8210 supports a full implementation of OSPF V2. For details on the IBM 8210 IP implementation, see B.5.4, "IBM 8210 IP and OSPF" on page 505.

B.5.1 OSPF Terminology

OSPF uses a specific terminology that must be understood before the protocol can be described.

B.5.1.1 Areas

OSPF internetworks are organized into *areas*.

An OSPF area consists of a number of networks and routers that are logically grouped together. Areas may be defined on a per location or a per region basis, or they may be based on administrative boundaries.

All OSPF networks consist of at least one area, the backbone, plus as many additional areas as are demanded by network topology and other design criteria.

Within an OSPF area all routers maintain the same topology database, exchanging link-state information to maintain their synchronization. This ensures that all routers calculate the same network map for the area.

Information about networks outside an area is summarized by *area border* or *AS boundary routers* (see B.5.1.3, "Intra-area, Area Border and AS Boundary Routers" on page 490) and flooded into the area. Routers within an area have no knowledge of the topology of networks outside the area, only of routes to destinations provided by area border and AS boundary routers.

The importance of the area concept is that it limits the size of the topology database that must be held by routers. This has a direct impact on the processing to be carried out by each router, and on the amount of link-state information that must be flooded into individual networks.

B.5.1.2 The OSPF Backbone

All OSPF networks must contain at least one area, the *backbone*, which is assigned an area identifier of *0.0.0.0*.

The backbone has all the properties of an area, but has the additional responsibility of distributing routing information between areas attached to it.

When routing a packet between two areas the backbone is used. The path that the packet will travel can be broken up into three contiguous pieces: an intra-area path from the source to an area router, a backbone path between source and destination area, and then another intra-area path to the destination. The OSPF algorithm finds a set of such paths that have the smallest cost.

Normally an OSPF backbone should be contiguous, that is, with all backbone routers attached to one another. This may not be possible because of network topology, in which case backbone continuity must be maintained by the use of *virtual links*.

Virtual links are backbone router-to-backbone router connections that traverse a non-backbone area.

Routers within the backbone operate identically to other intra-area routers and maintain full topology databases for the backbone area.

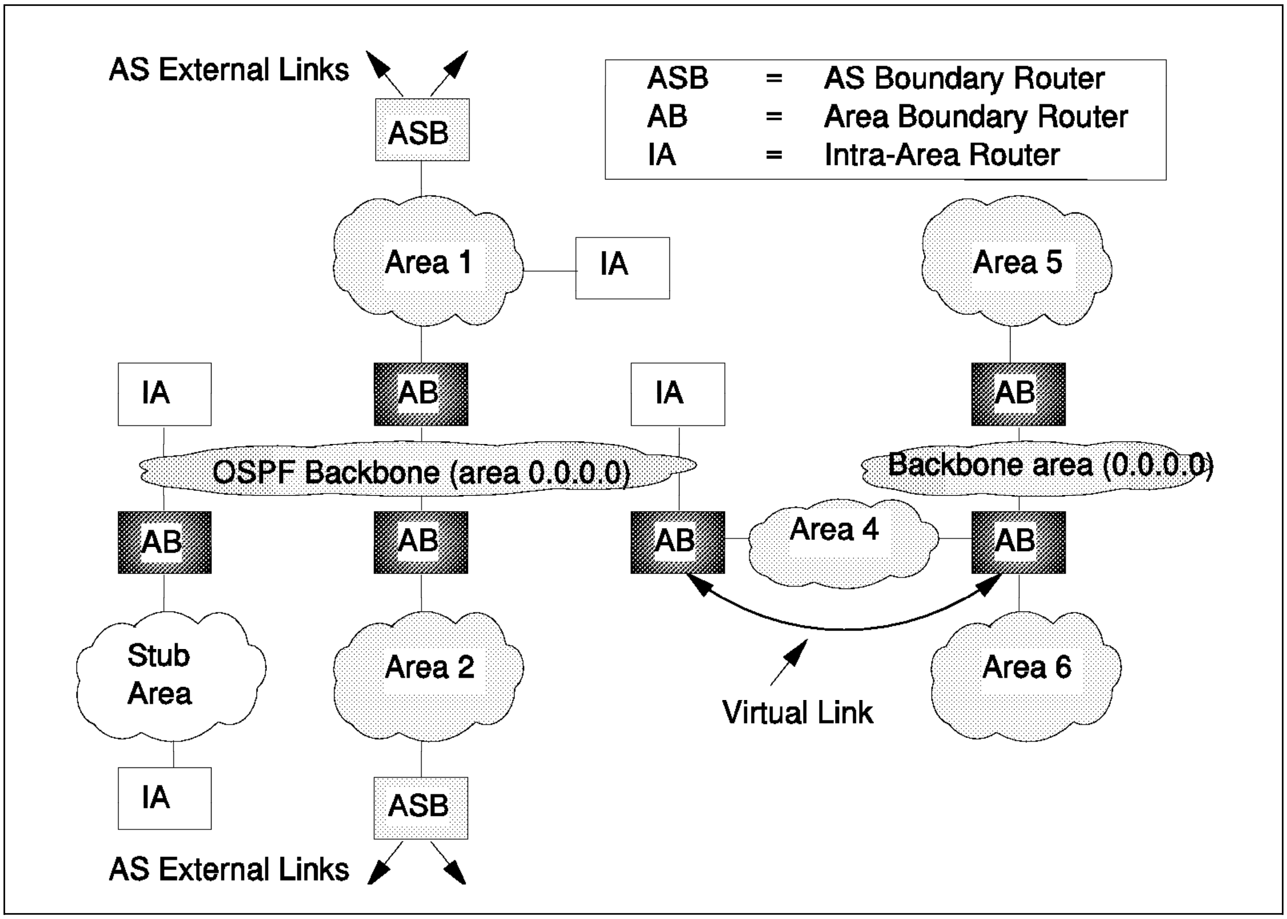


Figure 357. OSPF Network

B.5.1.3 Intra-area, Area Border and AS Boundary Routers

There are three possible types of routers in an OSPF network.

Routers that are situated entirely within an OSPF area are called *intra-area routers*. All intra-area routers flood router links advertisements into the area to define the links they are attached to. If elected designated or backup designated router (see B.5.1.6, "Designated and Backup Designated Router" on page 491) they also flood network links advertisements to define the identity of all routers attached to the network. Intra-area routers maintain a topology database for the area in which they are situated.

Routers that connect two or more areas are referred to as *area border routers*. Area border routers maintain topology databases for each area to which they are attached, and exchange link-state information with other routers in those areas. Area border routers also flood summary link-state advertisements into each area to inform them of inter-area routes.

Routers that are situated at the periphery of an OSPF internetwork and exchange reachability information with routers in other ASs using exterior gateway protocols are called *AS boundary routers*. Routers that import static routes or routes from other IGPs such as RIP into an OSPF network are also AS boundary routers. AS boundary routers are responsible for flooding AS external link-state advertisements into all areas within the AS to inform them of external routes.

Figure 357 shows the location of intra-area, area border and AS boundary routers within an OSPF internetwork.

B.5.1.4 Neighbor Router

Two routers that have interfaces to a common network are said to be *neighbors*.

Neighbor routers are discovered by the OSPF Hello protocol, which is described in B.5.2.1, "Discovering Neighbors - the OSPF Hello Protocol" on page 494.

B.5.1.5 Adjacent Router

Neighbor routers may become *adjacent*. They are said to be adjacent when they have synchronized their topology databases through the exchange of link-state information.

Link-state information is exchanged only between adjacent routers, not between neighbor routers.

Not all neighbor routers become adjacent. Neighbors on point-to-point links do so, but on multi-access networks adjacencies are only formed between individual routers and the designated and backup designated routers.

The exchange of link-state information between neighbors can create significant amounts of network traffic. Limiting the number of adjacencies on multi-access networks in this way achieves considerable reductions in network traffic.

B.5.1.6 Designated and Backup Designated Router

All multi-access networks have a *designated* and a *backup designated* router.

These routers are elected automatically for each network once neighbor routers have been discovered by the Hello protocol.

The designated router performs two key roles for a network:

- It generates network links advertisements that list the routers attached to a multi-access network.
- It forms adjacencies with all routers on a multi-access network and, therefore, becomes the focal point for forwarding of all link-state advertisements.

The backup designated router forms the same adjacencies as the designated router. It therefore has the same topology database and is able to assume designated router functions should it detect that the designated router has failed.

B.5.1.7 Stub Areas

In some ASs, the majority of the topological database may contain routes to destinations outside the AS. OSPF allows certain areas to be configured as *stub area*. OSPF external advertisements are not flooded into/throughout stub areas; routing to AS external destinations in these areas is based on default routing. This reduces the memory requirements for stub area′s internal routers.

On one or more of the stub area border routers a default route must be defined, which is then advertised into the stub area.

There are a couple of restrictions on the use of stub areas. Virtual links cannot be configured through stub areas. In addition, AS boundary routers cannot be placed internal to stub areas.

B.5.1.8 Point-to-Point and Multi-access Networks

All OSPF areas consist of aggregates of networks linked by routers. OSPF categorizes networks into two different types.

Point-to-point networks directly link two routers. OSPF packets on a point-to-point network are multicast to the neighbor router. *Multicasting* is the term used for transmitting IP datagrams to a functional rather than a specific IP address. A functional address will typically be recognized by a number of systems and can be considered a form of limited broadcast. OSPF defines the use of two multicast addresses (224.0.0.4 and 224.0.0.5) for OSPF router interactions.

Multi-access networks are those which support the attachment of more than two routers. They are further subdivided into two types:

- Broadcast
- Non-broadcast

Broadcast networks have the capability of directing OSPF packets to all attached routers, using an address that is recognized by all of them. A token-ring LAN is an example of a broadcast multi-access network.

Non-broadcast networks do not have this capability and all packets must be specifically addressed to routers on the network. This requires that routers on a non-broadcast network know the addresses of neighbors. A frame relay network is an example of a non-broadcast multi-access network.

B.5.1.9 Link-State Advertisements

Link-state information is exchanged by adjacent OSPF routers to allow area topology databases to be maintained, and inter-area and inter-AS routes to be advertised.

Link-state information consists of five types of *link-state advertisement*. Together these provide all the information needed to describe an OSPF network and:

1. Router links
2. Network links
3. Summary links (Type 3 and 4)
4. AS external links

Router links advertisements are generated by all OSPF routers and describe the state of the router′s interfaces (links) within the area. They are flooded throughout a single area only.

Network links advertisements are generated by the designated router on a multi-access network and list the routers connected to the network. They are flooded throughout a single area only.

Summary links advertisements are generated by area border routers. There are two types; one describes routes to destinations in other areas, and the other routes to AS boundary routers. They are flooded throughout a single area only.

AS external links advertisements are generated by AS boundary routers and describe routes to destinations external to the OSPF network. They are flooded throughout all areas in the OSPF network, except stub areas.

B.5.2 Protocol Description

The OSPF protocol is an implementation of a *link-state* routing protocol, as described in B.3.3, “Link-State Routing” on page 484.

OSPF packets are transmitted directly in IP datagrams. IP datagrams containing OSPF packets can be distinguished by their use of *protocol identifier 89* in the IP header. Therefore, OSPF packets are not contained in TCP or UDP headers. OSPF packets are always sent with IP *type of service* set to *0*, and the IP *precedence field* set to internetwork control. This is to aid them in getting preference over normal IP traffic.

Further details of IP protocol identifiers, type of service and precedence can be found in RFC 791 - *Internet Protocol*.

The IP destination address for OSPF packets is selected as follows. On physical point-to-point networks, the IP destination is always set to the address AllSPFRouters (that is, 224.0.0.5). On all other network types (including virtual links), the majority of OSPF packets are sent as unicasts. In this case, the IP destination is just the neighbor IP address. The only packets not sent as unicasts are on broadcast networks; on these networks Hello packets are sent to the multicast destination AllSPFRouters, the Designated Router and its Backup send both Link-State Update Packets and Link-State Acknowledgment Packets to the multicast address AllSPFRouters, while all other routers send both their Link-State Update and Link-State Acknowledgment packets to the multicast address AllDRouters (that is, 224.0.0.4).

Retransmissions of Link-State Update packets are always sent as unicasts.

The IP source address is the IP address of the sending interface. As for unnumbered point-to-point, no IP address is defined, and another IP address belonging to the router is used. On the 8210, the *router_ID* is used.

All OSPF packets share a common header, which is shown in Figure 358 on page 494.

This header provides general information, such as area identifier and originating router identifier, and also includes a checksum and authentication information. A type field defines each OSPF packet as one of five possible types:

1. Hello
2. Database Description
3. Link-State Request
4. Link-State Update
5. Link-State Acknowledgement

The router identifier, area identifier, and authentication information are configurable for each OSPF router.

The OSPF protocol defines a number of stages that must be executed by individual routers. They are as follows:

- Discovering neighbors
- Electing the designated router
- Initializing neighbors

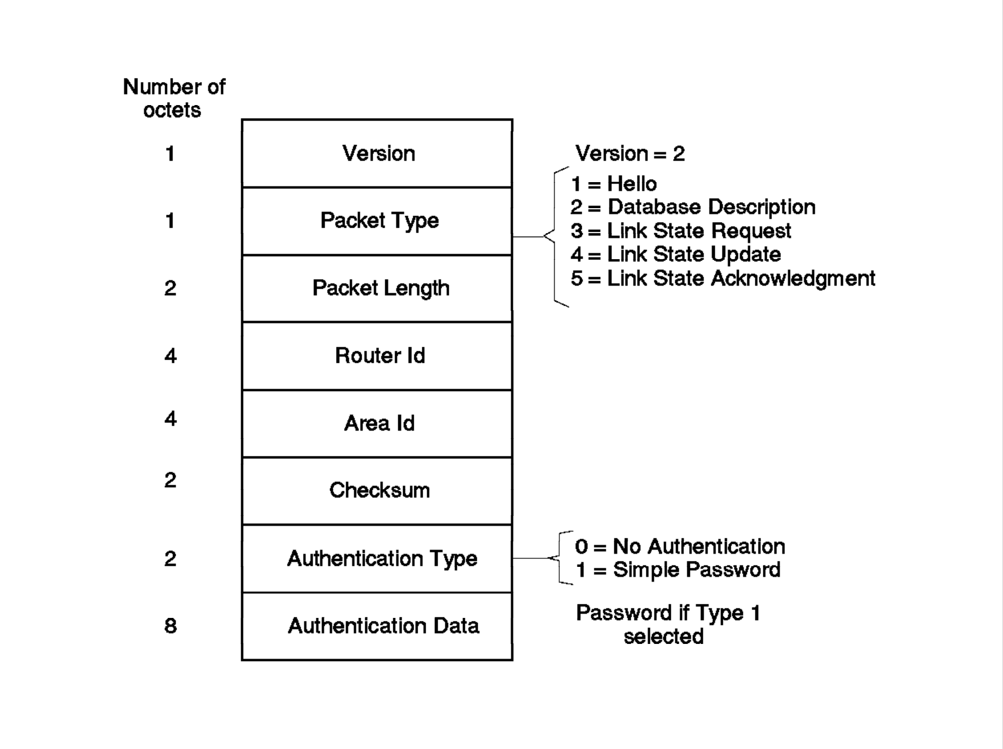


Figure 358. OSPF Common Header

- Propagating link-state information
- Calculating routing tables

The use of the five OSPF packet types to implement stages of the OSPF protocol are described in the following subsections.

During OSPF operation a router cycles each of its interfaces through a number of *states* from *Down*, through *Waiting*, to *DR Other*, *BackupDR* or *DR* (DR stands for *designated router*) depending on the status of each attached network and the identity of the designated router elected for each of them. A detailed description of these states is outside the scope of this document, but can be found in RFC 1583.

At the same time a router cycles each neighbor interface (interaction) through a number of states as it discovers them and then becomes adjacent. These states are *Down*, *Attempt*, *Init*, *2-Way*, *ExStart*, *Exchange*, *Loading* and *Full*. Once again a description of these is outside the scope of this document but can be found in RFC 1583.

B.5.2.1 Discovering Neighbors - the OSPF Hello Protocol

The Hello protocol is responsible for discovering neighbor routers on a network and establishing and maintaining relationships with them.

Hello packets are sent out periodically on all router interfaces. The format of these is shown in Figure 359 on page 495.

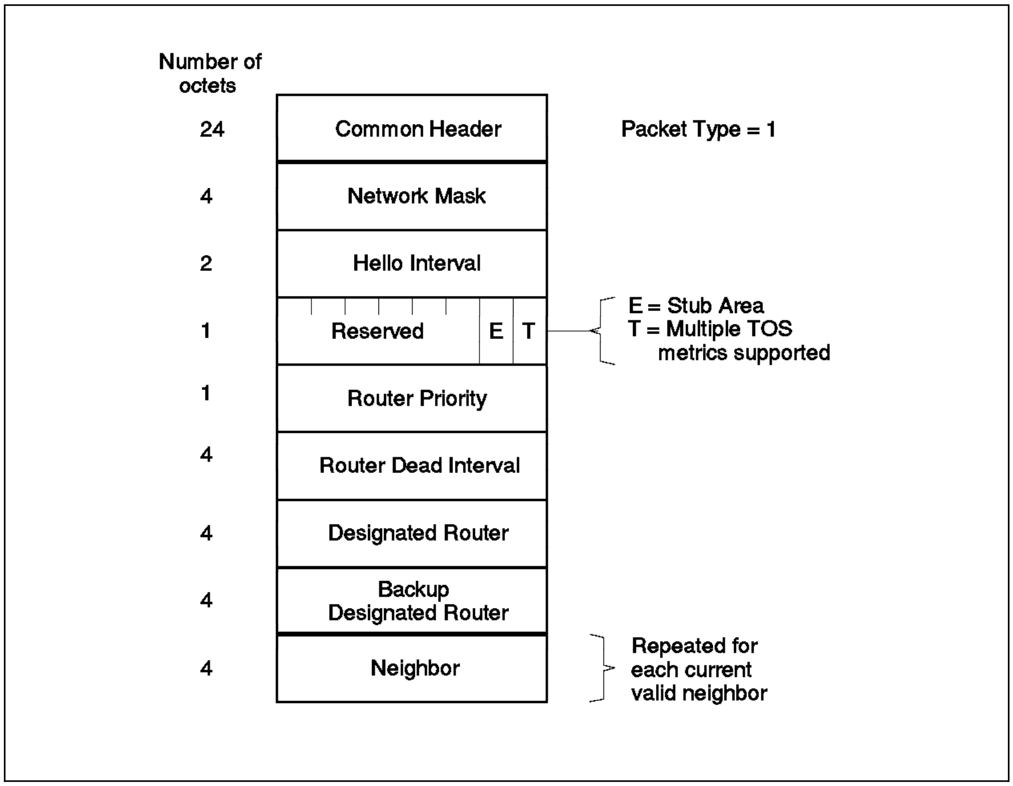


Figure 359. OSPF Hello Packet

Hello packets contain the identities of neighbor routers whose Hello packets have already been received over a specific interface. They also contain the *network mask*, *router priority*, *designated router identifier* and *backup designated router identifier*. The final three parameters are used to elect the designated router on multi-access networks.

The network mask, router priority, Hello interval and router dead interval are configurable for each interface on an OSPF router.

A router interface changes state from *Down* to *Point-to-Point* (if the network is point-to-point), to *DR Other* (if the router is ineligible to become designated router), or otherwise to *Waiting* as soon as Hello packets are sent over it.

A router receives Hello packets from neighbor routers via its network interfaces. When this happens the neighbor interface state changes from *Down* to *Init*. Bidirectional communication is established between neighbors when a router sees itself listed in a Hello packet received from another router. Only at this point are the two routers defined as true neighbors, and the neighbor interface changes state from *Init* to *2-Way*.

B.5.2.2 Electing the Designated Router

All multi-access networks have a designated router. There is also a backup designated router that takes over in the event that the designated router fails.

The use of a backup, which maintains an identical set of adjacencies and an identical topology database to the designated router, ensures there is no extended loss of routing capability if the designated router fails.

The designated router performs two major functions on a network:

- It originates network links advertisements on behalf of the network.
- It establishes adjacencies with all other routers on the network. Only routers with adjacencies exchange link-state information and synchronize their databases.

The designated router and backup designated router are elected on the basis of the *router identifier*, *router priority*, *designated router* and *backup designated router* fields in Hello packets. Router priority is a single-byte field that defines the priority of a router on a network. The lower the value of the priority field the more likely the router is to become the designated router, hence the higher its priority. A zero value means the router is ineligible to become designated or backup designated router.

The process of designated router election is as follows:

1. The current values for designated router and backup designated router on the network are initialized to 0.0.0.0.
2. The current values for router identifier, router priority, designated router and backup designated router in Hello packets from neighbor routers are noted. Local router values are included.
3. Backup Designated Router Election:

   Routers that have been declared as designateds router are ineligible to become backup designated routers.

   The backup designated router will be declared to be:

   - The highest priority router that has been declared as backup designated router
   - The highest priority router if no backup designated router has been declared

   If equal priority routers are eligible, the one with the highest router identifier is chosen.
4. Designated Router Election:

   The designated router will be declared to be:

   - The highest priority router that has been declared designated router
   - The highest priority router if no designated router has been declared
5. If the router carrying out the determination is declared the designated or backup designated router, then the previous steps are re-executed. This ensures that no router can declare itself both designated and backup designated router.

Once designated and backup designated routers have been elected for a network, they proceed to establish adjacencies with all routers on the network.

Completion of the election process for a network causes the router interface to change state from *Waiting* to *DR*, *BackupDR*, or *DR Other* depending on whether the router is elected the designated router, the backup designated router or neither of these.

B.5.2.3 Establishing Adjacencies - Database Exchange

A router establishes adjacencies with a subset of neighbor routers on a network.

Routers connected by point-to-point networks and virtual links always become *adjacent*. Routers on multi-access networks form adjacencies with the designated and backup designated routers only.

Link-state information flows only between adjacent routers. Before this can happen it is necessary for them to have the same topological database and to be synchronized.

This is achieved in OSPF by a process called *database exchange*.

Database exchange between two neighboring routers occurs as soon as they attempt to bring up an adjacency. It consists of the exchange of a number of database description packets that define the set of link-state information present in the database of each router. The link-state information in the database is defined by the list of link-state headers for all link-state advertisement in the database (see Figure 364 on page 500 for information on the link-state header).

The format of database description packets is shown in Figure 360.

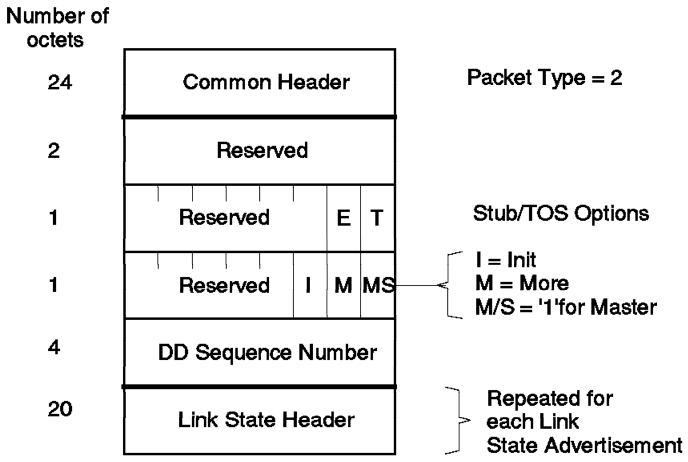


Figure 360. OSPF Database Description Packet

During the database exchange process the routers form a *master/slave* relationship, with the master being the first to transmit. The master sends database description packets to the slave to describe its database of link-state information. Each packet is identified by a sequence number and contains a list of the link-state headers in the master′s database. The slave acknowledges each packet by sequence number and includes its own database of headers in the acknowledgements.

Flags in database description packets indicate whether they are from a master or slave (the *M/S* bit), the first such packet (the *I* bit) and if there are more packets to come (the *M* bit). Database exchange is complete when a router receives a database description packet from its neighbor with the M bit off.

During database exchange each router makes a list of the link-state advertisements for which the adjacent neighbor has a more up to date instance (all advertisements are sequenced and time stamped). Once the process is complete each router requests these more up to date instances of advertisements using link-state requests.

The format of link-state request packets is shown in Figure 361.

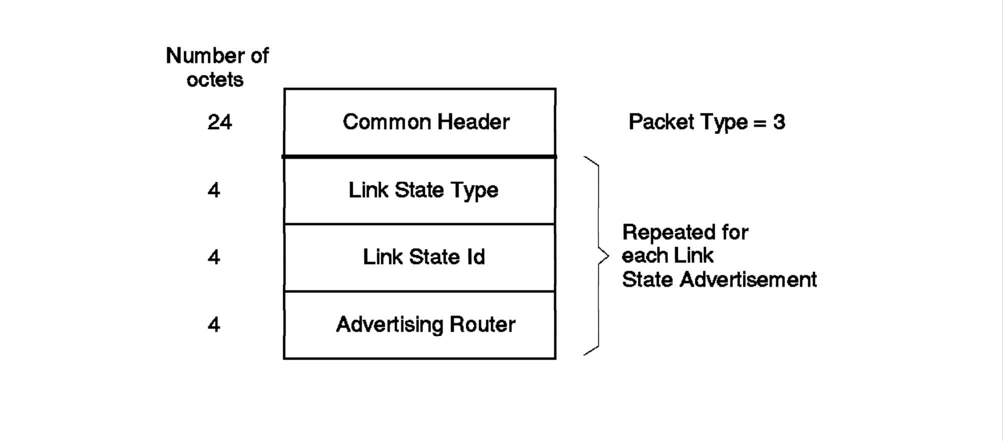


Figure 361. OSPF Link-State Request Packet

The database exchange process sequences the neighbor interface state from *2-Way* through:

ExStart as the adjacency is created and the master agreed upon

Exchange as the topology databases are being described

Loading as the link-state requests are being sent and responded to

Full when the neighbors are fully adjacent

In this way the two routers synchronize their topology databases and are able to calculate identical network maps for their OSPF area.

B.5.2.4 Link-State Propagation

Information about the topology of an OSPF network is passed from router to router in link-state advertisements.

Link-state advertisements pass between adjacent routers in the form of *link-state update* packets, the format of which is shown in Figure 362.

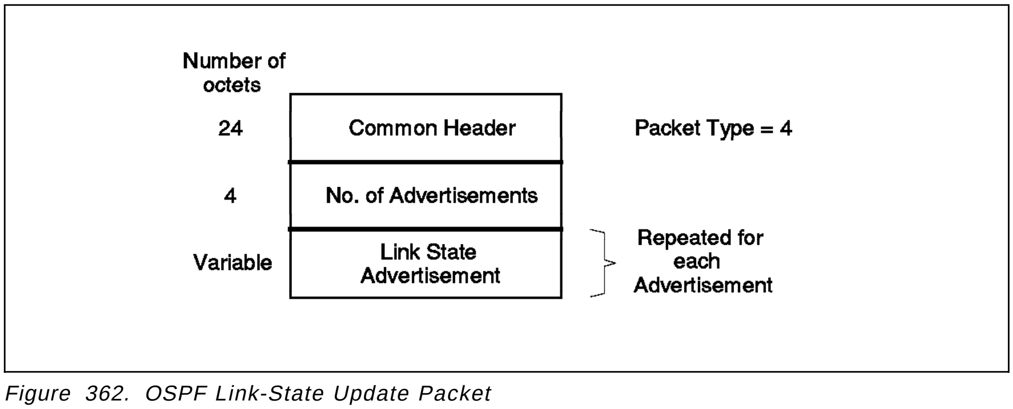


Figure 362. OSPF Link-State Update Packet

Link-state advertisements are of five types: router links, network links, summary links (two types) and AS external links as noted earlier in this section.

Link-state updates pass as a result of link-state requests during database exchange, and also in the normal course of events when routers wish to indicate a change of network topology. Individual link-state update packets can contain multiple link-state advertisements.

It is essential that each OSPF router in an area has the same network topology database, and hence the integrity of link-state information must be maintained.

For that reason link-state update packets must be passed without loss or corruption throughout an area. The process by which this is done is called *flooding*.

A link-state update packet floods one or more link-state advertisements one hop further away from their originator. To make the flooding procedure reliable each link-state advertisement must be acknowledged separately. Multiple acknowledgements can be grouped together into a single *link-state acknowledgement packet*. The format of the link-state acknowledgement packet is shown in Figure 363.

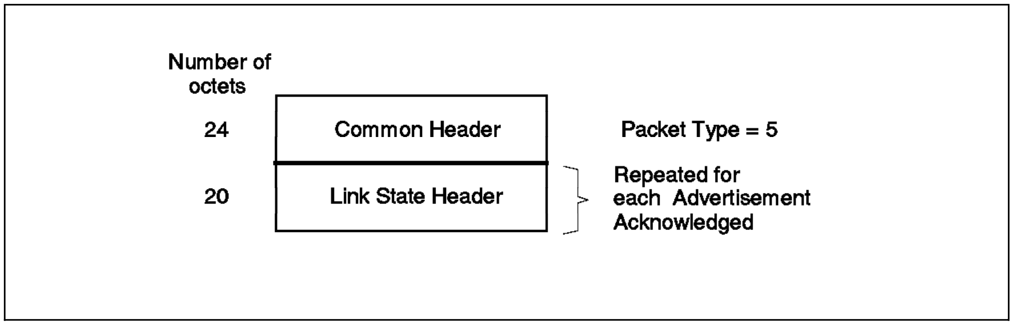


Figure 363. OSPF Link-State Acknowledgement Packet

In order to maintain database integrity it is essential that all link-state advertisements are rigorously checked to ensure validity.

The following checks are applied and the advertisement discarded if:

- The link-state checksum is incorrect
- The link-state type is invalid
- The advertisement′s age has reached its maximum
- The advertisement is older than or the same as one already in the database

If an advertisement passes the previous checks, then an acknowledgement is sent back to the originator. If no acknowledgement is received by the originator, then the original link-state update packet is retransmitted after a timer has expired.

Once accepted an advertisement is flooded onward over the router′s other interfaces until it has been received by all routers within an area.

Advertisements are identified by their *link-state type*, *link-state ID* and the *advertising router*. They are further qualified by their *link-state sequence number*, *link state age* and *link-state checksum number*.

The age of a link-state advertisement must be calculated to determine if it should be installed into a router′s database. Only a more recent advertisement should be accepted and installed. Advertisements are only considered more recent if they have a newer sequence number, if they have the larger checksum (if sequence numbers are equal), or if they have their age set to *max age* (if checksums are equal).

Valid link-state advertisements are installed into the topology database of the router. This causes the topology map or graph to be recalculated and the routing table to be updated.

Link-state advertisements all have a common header. This is shown in Figure 364. The five link-state advertisement types are shown in Figure 365 on page 501, in Figure 366 on page 501, in Figure 367 on page 502, and in Figure 368 on page 502.

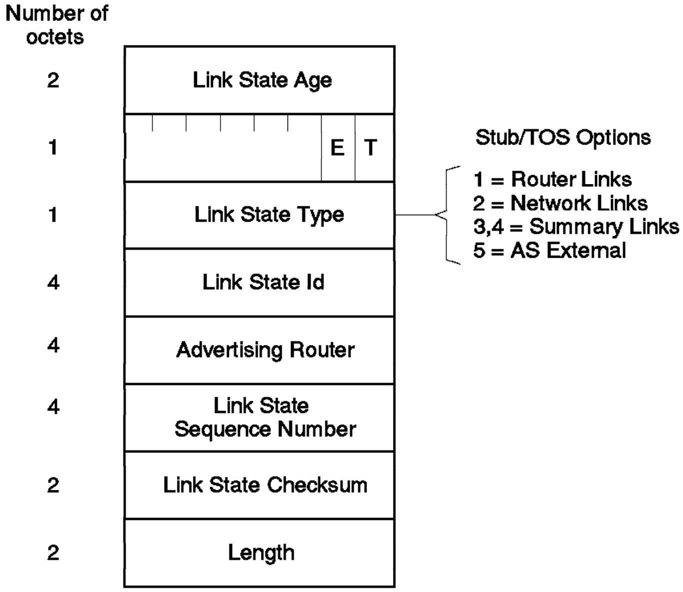


Figure 364. OSPF Link-State Header

B.5.2.5 Routing Table Calculation

Each router in an OSPF area builds up a topology database of validated link-state advertisements and uses them to calculate the network map for the area. From this map the router is able to determine the best route for each destination and insert it into its routing table.

Each advertisement contains an age field that is incremented while the advertisement is held in the database. An advertisement′s age is never incremented past *max age*. When max age is reached it is excluded from routing table calculation and reflooded through the area as a newly originated advertisement.

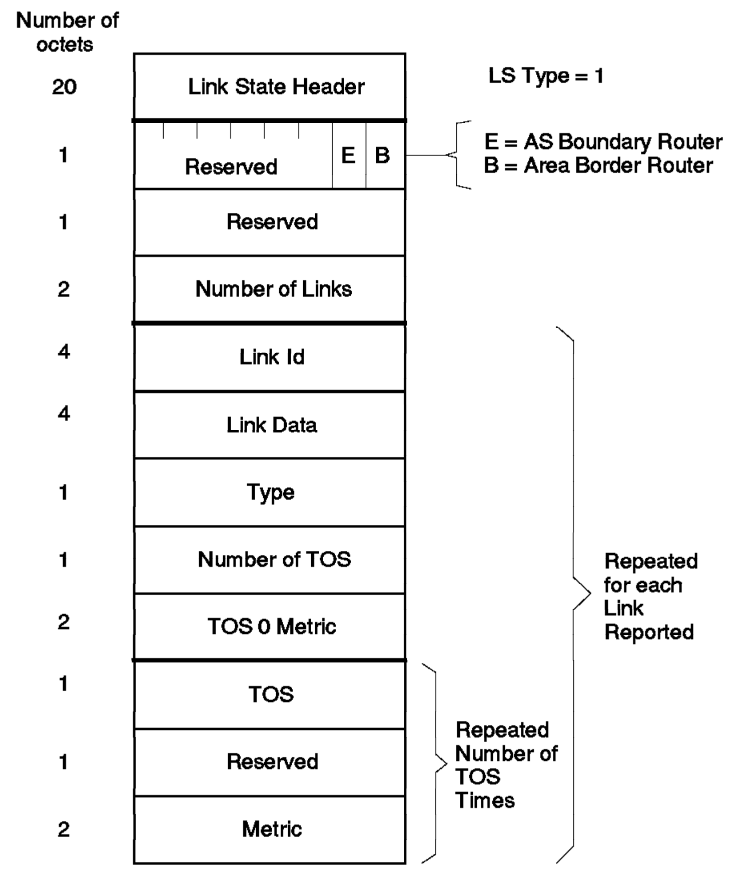


Figure 365. OSPF Router Links Advertisement

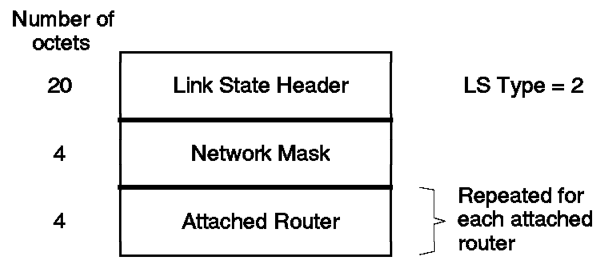


Figure 366. OSPF Network Links Advertisement

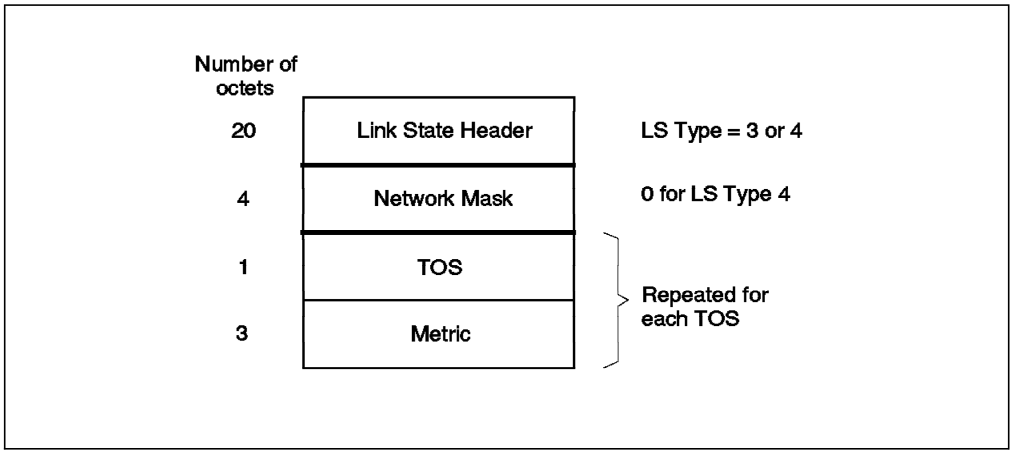


Figure 367. OSPF Summary Links Advertisement

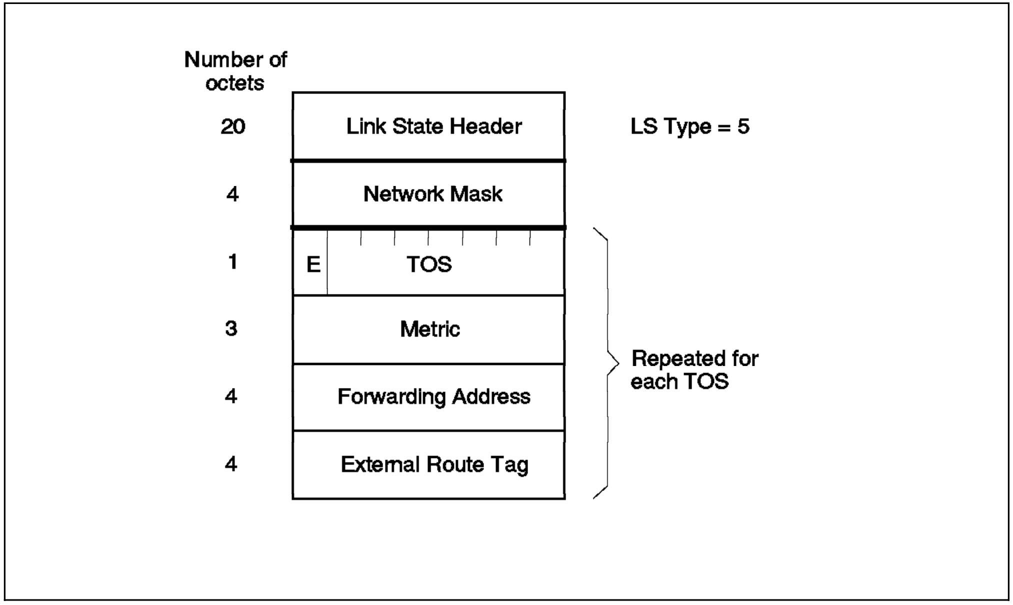


Figure 368. OSPF External Links Advertisement

Routers build up their routing table from the database of link-state advertisements in the following sequence:

1. The shortest path tree is calculated from router and network links advertisements allowing the best routes within the area to be determined.
2. Inter-area routes are added by examination of summary links advertisements.
3. AS external routes are added by examination of AS external links advertisements.

The topology graph or map constructed from the previous process is used to update the routing table. The routing table is recalculated each time a new advertisement is received.

B.5.3 Multicast Extensions to OSPF (MOSPF)

RFC 1584 describes an extension to OSPF Version 2, so that multicast routing capabilities can be added into an OSPF Version 2 routing domain.

IP multicasting is an extension of LAN multicasting to a TCP/IP internet. Multicasting support for TCP/IP hosts has been specified in RFC 1112. In that document, multicast groups are represented by IP class D addresses. Individual TCP/IP hosts join (and leave) multicast groups through the Internet Group Management Protocol (IGMP, also specified in RFC 1112). A host need not be a member of a multicast group in order to send datagrams to the group. Multicast datagrams are to be delivered to each member of the multicast group with the same *best-effort* delivery according to regular (unicast) IP data traffic.

MOSPF provides the ability to forward multicast datagrams from one IP network to another (that is, through Internet routers). MOSPF forwards a multicast datagram on the basis of both the datagram′s source and destination (this is sometimes called source/destination routing). The OSPF link-state database provides a complete description of the Autonomous System′s topology. By adding a new link-state advertisement (type 6), the *group-membership-LSA*, the location of all multicast group members, is pinpointed in the database. The path of a multicast datagram can then be calculated by building a shortest-path tree rooted at the datagram′s source. All branches not containing multicast members are pruned from the tree. These pruned shortest-path trees are initially built when the first datagram is received (that is, on demand). The results of the shortest path calculation are then cached for use by subsequent datagrams having the same source and destination.

Routers running MOSPF can be intermixed with non-multicast OSPF routers. Both types of routers can interoperate when forwarding regular (unicast) IP data traffic. Obviously, the forwarding extent of IP multicasts is limited by the number of MOSPF routers present in the Autonomous System (and their interconnection, if any). An ability to *tunnel* multicast datagrams through non-multicast routers is not provided. In MOSPF, just as in the base OSPF protocol, datagrams (multicast or unicast) are routed *as is*; they are not further encapsulated or decapsulated as they transit the Autonomous System.

B.5.3.1 IGMP Interface: The Local Group Database

The local group database keeps track of the group membership of the router′s directly attached networks. Each entry in the local group database is a (group, attached network) pair, which indicates that the attached network has one or more IP hosts belonging to the IP multicast destination group. This information is then used by the router when deciding which directly attached networks to forward a received IP multicast datagram onto, in order to complete delivery of the datagram to (local) group members.

The local group database is built through the operation of IGMP. When a MOSPF router becomes a designated router on an attached network, it starts sending periodic IGMP Host Membership Queries on the network. Hosts then respond with IGMP Host Membership Reports, one for each multicast group to which they belong. Upon receiving a Host Membership Report for a multicast group, the router updates its local group database by adding/refreshing the entry. If at a later time reports for a group cease to be heard on the network, the entry is then deleted from the local group database.

It is important to note that on any particular network, the sending of IGMP Host Membership Queries and the listening to IGMP Host Membership Reports is performed solely by the designated router. A MOSPF router ignores Host Membership Reports received on those networks where the router has not been elected designated router. This means that at most one router performs these IGMP functions on any particular network and ensures that the network appears in the local group database of at most one router. This prevents multicast datagrams from being replicated as they are delivered to local group members. As a result, each router in the Autonomous System has a different local group database.

B.5.3.2 Inter-Area Multicasting

Group-membership-LSAs are specific to a single OSPF area. This means that, just as with OSPF router-LSAs, network-LSAs and summary-link-LSAs, a group-membership-LSA is flooded throughout a single area only. A router attached to multiple areas may end up originating several group-membership-LSAs concerning a single multicast destination, one for each attached area.

Just as in OSPF, each MOSPF area has its own link-state database. The MOSPF database is simply the OSPF link-state database enhanced by the group-membership-LSAs.

B.5.3.3 Intra-Area Multicasting

In OSPF, the area border routers forward routing information and data traffic between areas. In MOSPF, a subset of the area border routers, called the *inter-area multicast forwarders*, forward group membership information and multicast datagrams between areas. Whether a given OSPF area border router is also an MOSPF inter-area multicast forwarder is configuration-dependent.

In order to convey group membership information between areas, inter-area multicast forwarders *summarize* their attached areas group membership to the backbone. This is a very similar functionality to the summary-link-LSAs that are generated in the base OSPF protocol. An inter-area multicast forwarder calculates which groups have members in its attached non-backbone areas. Then for each of these groups, the inter-area multicast forwarder injects a group-membership-LSA into the backbone area.

However, unlike the summarization of unicast destinations in the base OSPF protocol, the summarization of group membership in MOSPF is asymmetric. While a non-backbone area′s group membership is summarized to the backbone, this information is not then readvertised into other non-backbone areas. Nor is the backbone′s group membership summarized for the non-backbone areas.

To accomplish intra-area multicasting, the notion of *wild-card multicast receivers* is introduced. A wild-card multicast receiver is a router to which all multicast traffic, regardless of multicast destination, should be forwarded. A router′s wild-card multicast reception status is per-area. In non-backbone areas, all inter-area multicast forwarders are wild-card multicast receivers. This ensures that all multicast traffic originating in a non-backbone area will be forwarded to its inter-area multicast forwarders and, hence, to the backbone area. Since the backbone has complete knowledge of all areas′ group membership, the datagram can then be forwarded to all group members.

B.5.3.4 Interaction with the IGMP Protocol

MOSPF uses the IGMP protocol to monitor multicast group membership. In short, the designated router on a network periodically sends IGMP Host Membership Queries, which, in turn, elicit IGMP Host Membership Reports from the network′s multicast group members. These Host Membership Reports are then recorded in the designated router′s and backup designated router′s local group databases.

Only the network′s designated router sends Host Membership Queries. This minimizes the amount of group membership information on the network, both in terms of queries and responses.

Received Host Membership Reports are processed by both the network′s designated router and backup designated router. It is the designated router′s responsibility to distribute the network′s group membership information throughout the routing domain. The backup designated router processes reports so that it too has a complete picture of the network′s group membership, enabling a quick cutover upon designated router failure.

B.5.4 IBM 8210 IP and OSPF

The IBM 8210 IP router supports the following features of OSPF which have implementation specific behavior:

- Area border (AB) router support
- Support for stub areas
- Autonomous System border (ASB) support
- OSPF interface support
- Equal-cost multipath routing
- Simple authentication
- OSPF routing policies
- Multicast OSPF (MOSPF) support

There is no support for type of service (TOS) based routing, that is, TOS 0 is the only supported TOS.

For details on configuring IBM 8210 IP OSPF functions, see 7.3.5, “OSPF Configuration on the IBM 8210” on page 182.

B.5.4.1 Area Border (AB) Router Support

IBM 8210 IP supports attachment to multiple areas and summarization of routing information between areas. Area border routers must attach to the backbone (0.0.0.0) and at least one other area. Summarization information from one area will manifest itself as type 3 and 4 link-state advertisements (LSAs) in other areas.

B.5.4.2 Support for Stub Areas

IBM 8210 IP supports attachment to stub areas. OSPF Autonomous System external (ASE) LSAs will not be advertised into stub areas. Rather a type 3 network summary LSA for the default route (destination/mask 0.0.0.0) is generated.

B.5.4.3 Autonomous System Border (ASB) Support

IBM 8210 IP can be configured as an Autonomous System border (ASB) router. Non-OSPF routes can be imported into OSPF as OSPF Autonomous System externals (ASEs). This implies that the IBM 8210 IP will generate type 5 LSAs.

B.5.4.4 OSPF Interface Support

IBM 8210 supports the following types of OSPF interfaces:

- Numbered point-to-point (PtP)

  Numbered PtP connections are links to which an IP address has been assigned. Examples are ESCON and PPP connections.

- Unnumbered point-to-point (PtP)

  Unnumbered PtP connections are links to which no IP address has been assigned. OSPF packets will be sent using the IBM 8210 IP router-ID as source address.

- Broadcast

  An example is an emulated LAN interface. Link-level multicast is used to broadcast OSPF frames to all attached OSPF routers.

- Non-broadcast multi-access (NBMA)

  Viewing a network, for example, a frame relay network, as an NBMA network can be used when the network is fully meshed (meaning virtual circuits exist between any pair of routers). In routers that are eligible to become designated routers, neighbors must be configured as well whether or not the neighbor is eligible to become a designated router. The configurable poll interval defines the interval at which the designated router will attempt to contact the neighboring routers to establish an adjacency.

  NMBA connections are supported for Classical IP networks. If the network is partially meshed, it is more useful to view the network as a point-to-multipoint network.

- Point-to-multipoint (PtM)

  This interface type is used to allow NBMA topologies to be non-fully meshed. Rather than electing a designated router for the network and having that router generate network LSAs for the network, each router includes its neighbors in its router LSAs. When the route table is calculated, the network topology will appear as multiple point-to-point links rather than a single cloud. On one side of the frame relay virtual circuit, the neighboring router must be configured to allow the two routers to form an OSPF adjacency.

  Point-to-multipoint connections are supported for Classical IP networks that are not fully meshed.

- Virtual link

  Virtual links are supported to extend the backbone area′s connectivity through a transit area. The two end-points of the virtual link are area border routers (see B.5.1.2, “The OSPF Backbone” on page 489).

B.5.4.5 Equal-Cost Multipath Routing

IBM 8210 supports up to four equal-cost next hops for a route. When multiple next hops exist, the traffic to the destination is spread over the next hops round-robin.

B.5.4.6 Simple Password Authentication

IBM 8210 supports both simple password and no authentication. When simple password authentication is used, an 8-byte password is included in each OSPF packet. Upon reception, this password is validated with packets failing validation being dropped.

The authentication type (simple or none) is configured on the area level, while the authentication key is configured for each interface in areas with simple password authentication enabled.

B.5.4.7 OSPF Routing Policies

The IBM 8210 OSPF policy can be explained in terms of rules for:

- Advertisements of OSPF routes
- Import of external routes into OSPF as OSPF AS external (ASE) routes
- Generation of the default route and import as an OSPF ASE LSA
- OSPF route policy when multiple routes to a destination exist

OSPF Advertisement of Routes: OSPF allows filtering of LSAs on area boundaries only. All routers within an area have the same view of the area topology.

To limit the number of LSAs advertised outside the area, one can configure the network ranges associated with an area at the area boundary. This, in effect, will aggregate a number of networks into a single advertisement. The cost associated with the network range will be the lowest for any of the component networks. Additionally, one can define a network range that will not be visible.

Normally, OSPF ASE LSAs are flooded throughout the entire routing domain. One can prevent this for a given area by defining the area as a stub area. Within stub area, the area border router will advertise a single default route (destination/mask 0.0.0.0).

Importing Routes into OSPF: IBM 8210 allows configuration whether or not non-OSPF routes should be imported as OSPF AS external routes and advertised as OSPF ASE LSAs throughout the OSPF routing domain.

Imported routes can be imported as ASE type 1 (router) or ASE type 2 (network) routes. ASE type 1 routes always override type 2 routes. It has a single metric, which is the sum of the path cost to the AS border router, and the AS border router metric. Conversely, the metric for a type 2 ASE has an internal and external components: the path cost to the AS border router and the route′s external metric. When comparing ASE type 2 routes to the same destination, the one with the lower external metric will always be preferred, independent of the internal metric. The metric used for both the OSPF ASE type 1 route and the external component of the type 2 route is the metric from the protocol from which it is imported.

Default Route: IBM 8210 allows generation of default routes and advertisement of the routes through an AS external (ASE) LSA. For details, see 7.6, "Routing Protocols Interoperability" on page 197.

B.6 Border Gateway Protocol (BGP)

The *Border Gateway Protocol* (BGP) Version 4 is an exterior gateway protocol defined in RFC 1654. A companion document, RFC 1656 - *BGP-4 Protocol Document Roadmap and Implementation Experience*, details experience of its use in the Internet.

BGP was built upon experience with EGP and is intended as a functionally superior replacement for it. BGP Version 4 is an inter-Autonomous System gateway protocol. It has no ability to determine routes within an AS. It must rely on an interior gateway protocol, such as RIP or OSPF, to do this. It is an IAB standard protocol; its status is recommended.

BGP was introduced in the Internet for 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 multiple different routes into single routes. This significantly reduces the amount of routing information that BGP routers must store and exchange.

B.6.1 Introduction

BGP is not a routing protocol, but a reachability 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 addresses 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 its BGP neighbors (its clients). Each BGP router opens a connection and listens for incoming connections from neighbors at this well-known address. The router also opens active TCP connections to 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* connections, while connections between BGP speakers in different Autonomous Systems are *external* connections. A single AS may have one or many BGP connections to outside Autonomous Systems.

Figure 369 on page 509 shows three Autonomous Systems. The BGP speakers in AS-1 establish an internal connection with each other and an external connection with their neighbor in AS-2 and AS-3, respectively. Once the connections have been established, the routers will be able to share reachability information.

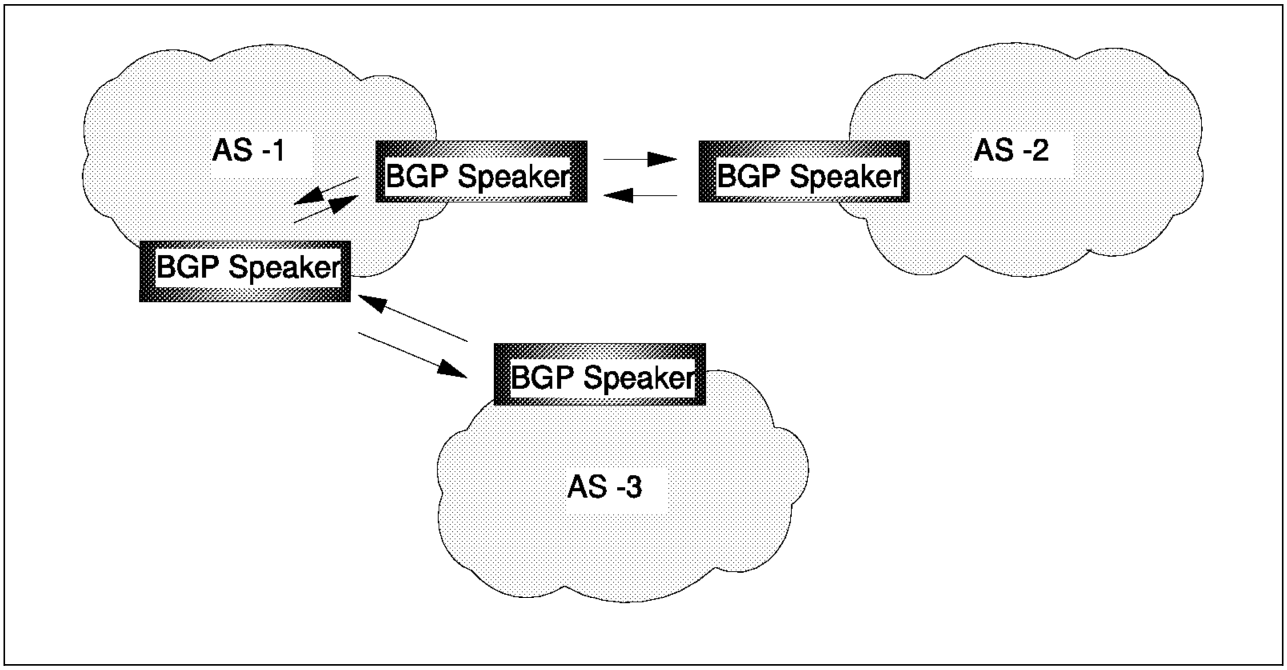


Figure 369. BGP Connections

B.6.1.1 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

  The originate policies decide which interior gateway protocol routes are advertised to BGP neighbors.

- Send Policies

  The send policies decide which BGP received routes are forwarded to BGP neighbors.

- Receive Policies

  The receive policies decide which BGP received routes are injected into the interior gateway protocol routing tables.

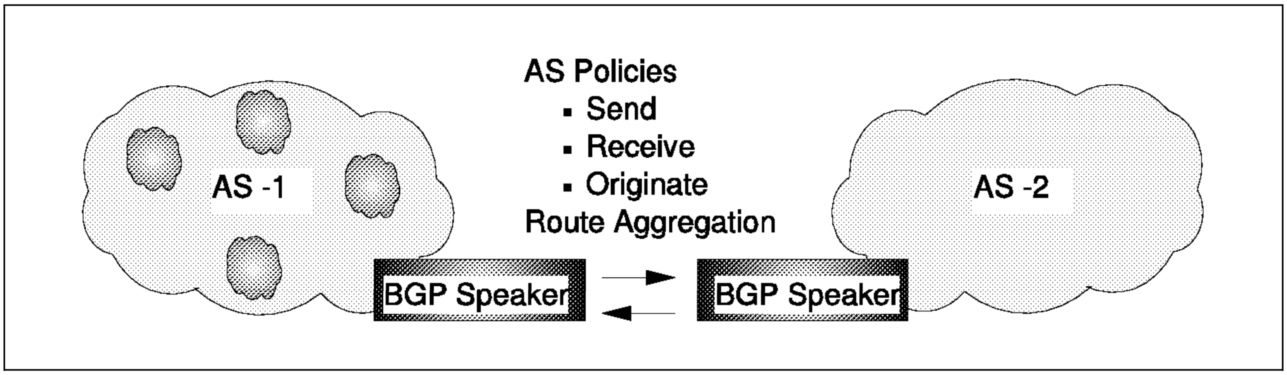


Figure 370. Route Aggregation and AS Policies

Once a TCP connection is established, the BGP speaker in AS-1, shown in Figure 370, can send its entire routing table to its BGP neighbor in AS-2. However, for security or other reasons, it may not be desirable to send reachability information on each network to AS-2. Similarly, it may not be desirable for AS-2 to receive reachability information on each network in AS-1.

In addition to deciding which routing information is sent, the BGP speaker also has the ability to condense its routing information. As an example, assume that AS-1 contains multiple class C networks. Instead of sending separate routes for each individual network, an *aggregated* route comprising all networks can be sent. For details, see B.6.2.4, "Route Aggregation" on page 515.

B.6.2 Protocol Description

The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems. This network reachability information includes information on the list of Autonomous Systems (ASs) that reachability information traverses. This information is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and some policy decisions at the AS level may be enforced.

BGP-4 provides a new set of mechanisms for supporting Classless Inter-Domain Routing. These mechanisms include support for advertising an IP prefix and eliminates the concept of network *class* within BGP. BGP-4 also introduces mechanisms that allow aggregation of routes, including aggregation of AS paths. These changes provide support for supernetting scheme.

BGP runs over a reliable transport protocol. This eliminates the need to implement explicit update fragmentation, retransmission, acknowledgement, and sequencing. Any authentication scheme used by the transport protocol may be used in addition to BGP′s own authentication mechanisms. The error notification mechanism used in BGP assumes that all outstanding data will be delivered before the connection is closed.

BGP uses TCP as its transport protocol. TCP meets BGP′s transport requirements and is present in virtually all commercial routers and hosts. BGP uses TCP port 179 for establishing its connections.

BGP systems form a transport protocol connection between one another. They exchange messages to open and confirm the connection parameters. The initial data flow is the entire BGP routing table. Incremental updates are sent as the routing tables change. BGP does not require periodic refresh of the entire BGP routing table. Therefore, a BGP speaker must retain the current version of the entire BGP routing tables of all of its peers for the duration of the connection. Keep alive messages are sent periodically to ensure the liveliness of the connection. Notification messages are sent in response to errors or special conditions. If a connection encounters an error condition, a notification message is sent and the connection is closed.

If a particular AS has multiple BGP speakers and is providing transit service for other ASs, then care must be taken to ensure a consistent view of routing within the AS. A consistent view of the interior routes of the AS is provided by the interior routing protocol. A consistent view of the routes exterior to the AS can be provided by having all BGP speakers within the AS maintain direct BGP connections with each other. Using a common set of policies, the BGP speakers arrive at an agreement as to which border routers will serve as exit/entry points for particular networks outside the AS. This information is communicated to the AS′s internal routers, possibly via the interior routing protocol. Care must be taken to ensure that the interior routers have all been updated with transit information before the BGP speakers announce to other ASs that transit service is being provided.

Connections between BGP speakers of different ASs are referred to as *external* links. BGP connections between BGP speakers within the same AS are referred to as *internal* links. Similarly, a peer in a different AS is referred to as an external peer, while a peer in the same AS may be described as an internal peer.

The BGP protocol comprises four main stages:

- Opening and confirming a BGP connection with a neighbor router
- Maintaining the BGP connection
- Sending reachability information
- Notification of error conditions

B.6.2.1 Opening and Confirming a BGP Connection

BGP communications between two routers commences with the TCP transport protocol connection being established. A description of this is outside the scope of this document, but can be found in *TCP/IP Tutorial and Technical Overview*, GG24-3376.

Once the connection is established, each router sends an *open* message to its neighbor.

The BGP open message, like all BGP messages, consists of a standard header plus packet type specific contents. The standard header consists of a 16-byte *marker* field, which is set to all ones, the *length* of the total BGP packet, and a *type* field that specifies the packet to be one of four possible types:

1. Open
2. Update
3. Notification
4. Keep alive

The format of the BGP header is shown in Figure 371 on page 512.

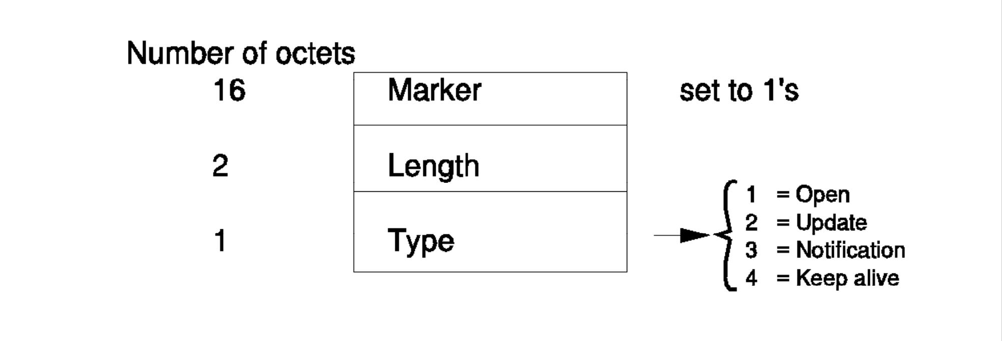


Figure 371. BGP Message Header

The open message defines the originating router′s *AS number*, its BGP *router identifier* and the *hold time* for the connection. If no keep alive, update or notification messages are received for a period of hold time, the originating router assumes an error, sends a notification message, and closes the connection.

The open message also provides an *authentication code* and *authentication data*. The use of these fields is not fully defined in the RFC and current BGP implementations use authentication code 0 with authentication data of all zeros.

The format of the open message is shown in Figure 372.

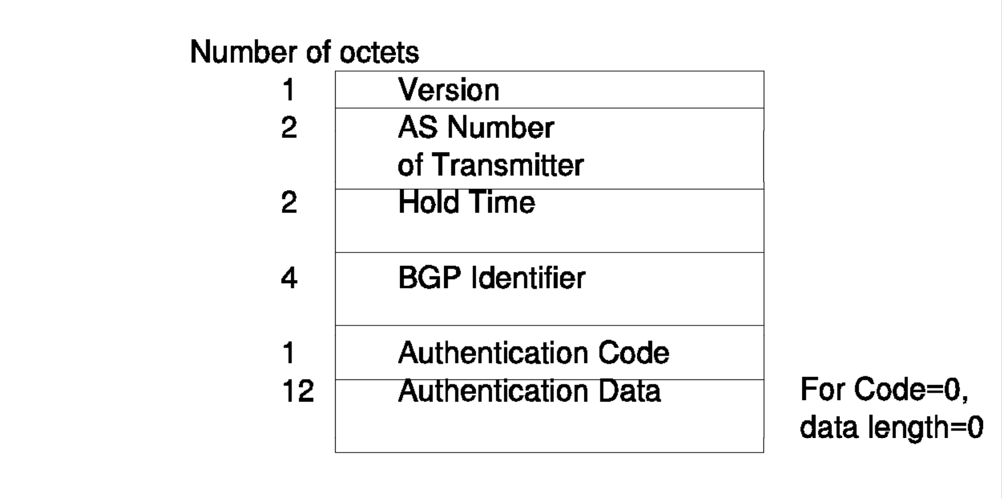


Figure 372. BGP Open Message

An acceptable open message is acknowledged by a *keep alive* message. Once neighbor routers have sent keep alives in response to opens, they can proceed to exchange further keep alives, notifications and updates.

B.6.2.2 Maintaining the BGP Connection

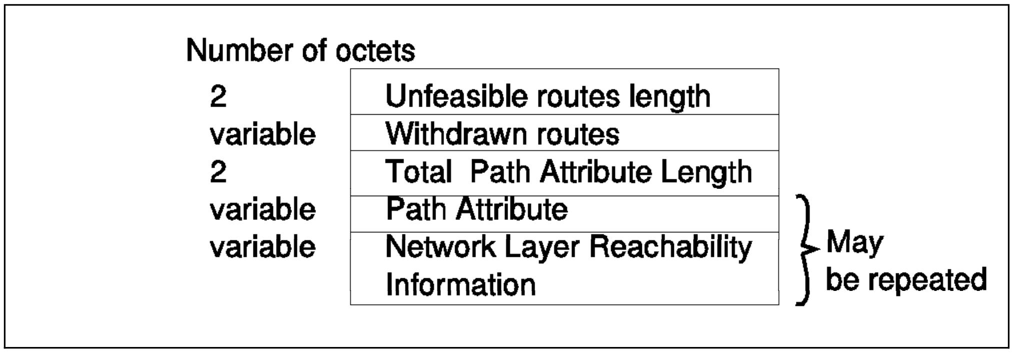
BGP messages must be exchanged periodically between neighbors. If no messages are received for a period defined by hold time in the open message, then an error on the connection is assumed.

BGP uses keep alive messages to maintain the connection between neighbors. Keep alive messages consist of the BGP packet header only with no data. The RFC recommends that they should be sent at intervals of approximately one third of hold time.

B.6.2.3 Sending Reachability Information

Reachability information is exchanged between BGP neighbors in update messages. Update messages are used to transfer routing information between BGP peers. The information in the update packet can be used to construct a graph describing the relationships of the various Autonomous Systems. By applying rules to be discussed, routing information loops and some other anomalies may be detected and removed from inter-AS routing.

An update message is used to advertise a single feasible route to a peer, or to withdraw multiple unfeasible routes from service. An update message may simultaneously advertise a feasible route and withdraw multiple unfeasible routes from service. The update message always includes the fixed-size BGP header and can optionally include the other fields as shown in Figure 373.

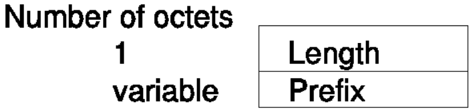


Number of octets	
2	Unfeasible routes length
variable	Withdrawn routes
2	Total Path Attribute Length
variable	Path Attribute
variable	Network Layer Reachability Information

Figure 373. BGP Update Message

The *unfeasible routes length* is a 2-octet field that indicates the total length of the withdrawn routes field. A value of 0 indicates that no routes are being withdrawn from service, and that the withdrawn routes field is not present in this update message.

The *withdrawn routes* field is a variable length field that contains a list of IP address prefixes for the routes that are being withdrawn from service. Each IP address prefix is encoded as a (length, prefix) pair.



Number of octets	
1	Length
variable	Prefix

> The length field indicates the length in bits of the IP address prefix. A length of zero indicates a prefix that matches all IP addresses. When the length field is zero, no prefix field is included.
>
> The prefix field contains IP address prefixes followed by enough trailing bits to make the end of the field fall on an octet boundary. Note that the value of trailing bits is irrelevant.

The *total path attribute length* is a 2-octet unsigned integer that indicates the total length of the path attributes field in octets. A value of 0 indicates that no network layer reachability information field is present in this update message.

Each *path attribute* consists of a triple set of values: *attribute flags*, *attribute type* and *attribute value*. Three of the attribute flags provide information about the status of the attributes types and may be *optional* or *well-known*, *transitive* or *non-transitive and partial* or *complete*.

Attribute flags must be read in conjunction with their associated attribute types. There are five attribute types which together define an advertised route.

The first three of these attributes are mandatory and must be supplied with all network advertisements. Each attribute type sent in a BGP update message must have its attribute flags set as indicated, otherwise an error occurs.

The format of BGP path attributes is shown in Figure 374.

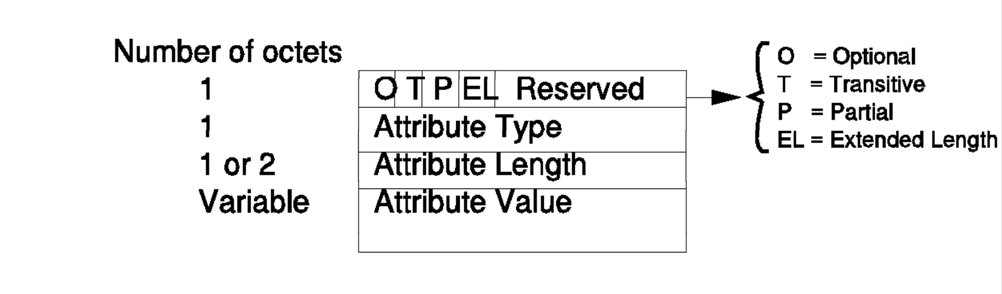


Figure 374. BGP Path Attributes

The remaining octets of the path attribute represent the attribute value and are interpreted according to the attribute flags and the attribute type code. The supported attribute type codes, their attribute values and uses are the following:

- Origin (Type Code 1)

  A well-known mandatory attribute that defines the origin of the route as an interior gateway protocol, an exterior gateway protocol or other (for example a static route).

- *AS_Path* (Type Code 2)

  AS_Path is a well-known mandatory attribute that is composed of a sequence of AS path segments. Each AS path segment is represented by a triple segment (path segment type, path segment length, and path segment value).

  > The *path segment* type is a 1-octet long field with the following values defined:
  >
  > – 1 - AS_SET, an unordered set of ASs a route in the update message has traversed
  >
  > – 2 - AS_SEQUENCE, an ordered set of ASs a route in the update message has traversed
  >
  > The *path segment* length is a 1-octet long field containing the number of ASs in the path segment value field.
  >
  > The *path segment* value field contains one or more AS numbers, each encoded as a 2-octet long field.

- Next_Hop (Type Code 3)

  This field contains the IP address of the border router that should be used as the next hop to the destinations listed in the Network Layer Reachability field of the update message.

- Multi_Exit_Disc (Type Code 4)

This is an optional nontransitive attribute that is a four-octet non-negative integer. The value of this attribute is used to discriminate among multiple exit points to a neighboring Autonomous System.

- Local_Pref (Type Code 5)

  Local_Pref is a well-known discretionary attribute that is a four-octet non-negative integer. It is used by a BGP speaker to inform other BGP speakers in its own Autonomous System of the originating speaker′s degree of preference for an advertised route.

- Atomic_aggregate (Type Code 6)

  Atomic_aggregate is a well-known discretionary attribute of length 0. It is used by a BGP speaker to inform other BGP speakers that the local system selected a less specific route without selecting a more specific route that is included in it.

- Aggregator (Type Code 7)

  Aggregator is an optional transitive attribute of length 6. The attribute contains the last AS number that formed the aggregate route (encoded as 2 octets), followed by the IP address of the BGP speaker that formed the aggregate route (encoded as 4 octets).

The *Network Layer Reachability Information* is a variable length field containing a list of IP address prefixes. Each IP address prefix is encoded as a (length, prefix) pair.

Number of octets	
1	Length
variable	Prefix

The Prefix field contains IP address prefixes and is followed by enough trailing bits to make the end of the field fall on an octet boundary.

An update message can advertise at most one route, which may be described by several path attributes. All path attributes contained in a given update message apply to the destinations carried in the Network Layer Reachability Information field of the update message.

B.6.2.4 Route Aggregation

Previous versions of BGP did not support supernetting nor subnetting as update messages being sent contained 32-bit Internet addresses that represented the (class A, B, or C) network that was reachable. That is, older BGP versions are *class-oriented*.

In BPG Version 4, however, networks are represented by a two-component structure (see, for example, the Network Layer Reachability Information field described in the previous section).

Number of octets	
1	Length
variable	Prefix

The Length field indicates the number of bits in the address prefix and is followed by the number of octets to hold the prefix.

When BPG speakers are exchanging routing information, they not only exchange routes but they also condense (that is *aggregate*) routes. Route aggregation leads to smaller routing tables.

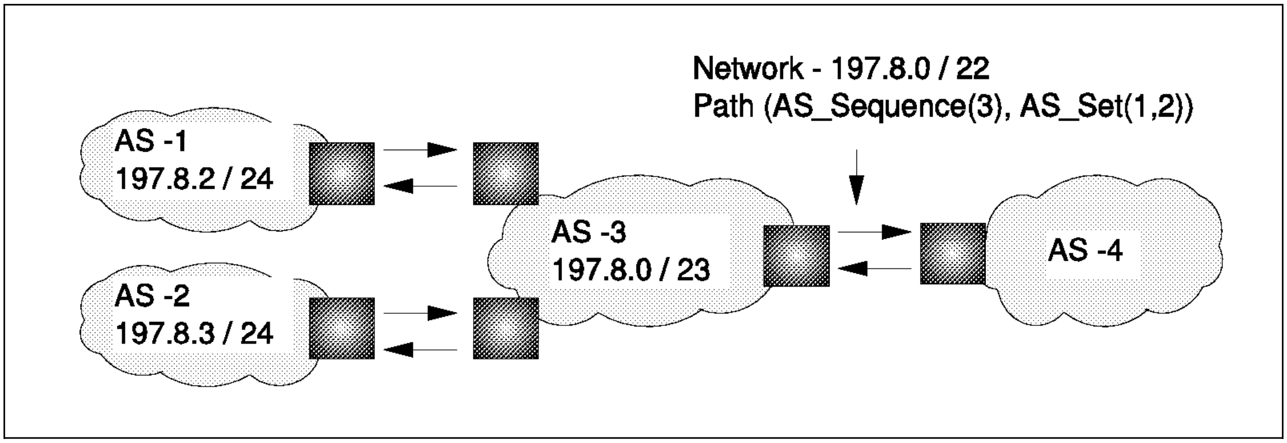


Figure 375. Route Aggregation

Figure 375 depicts an Internet consisting of five ASs. AS-1 and AS-2 each use a single class C subnet, 197.8.2. and 197.8.3, respectively. The routes from AS-3 to AS-1 and AS-2 can be represented as a 24-bit prefix of 197.8.2 and 197.8.3, respectively.

AS-3 uses two class C subnets, 197.8.0. and 197.8.1. The routes from AS-1 and AS-2 to AS-3 can be represented as a 23-bit prefix of 197.8.0. Hereby, two class C subnet routes are aggregated in a single route.

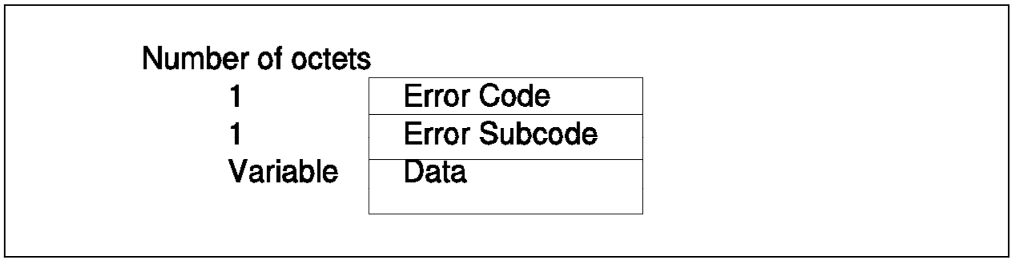
Maximum route aggregation can be accomplished when AS-3 informs AS4 about the routes to AS-3 itself, AS-1, and AS-2. Instead of announcing three paths, AS-3 forwards a route consisting of a 22-bit prefix of 197.8.0. In addition to the network information, path attributes are sent to indicate the numbers of the ASs that are reachable.

Note: As can be seen in this example, route aggregation strongly relies on allocating consecutive address ranges within ASs.

B.6.2.5 Notifying Errors

Notification messages are sent to a neighbor router when error conditions are detected. The BGP transport connection is closed immediately after a notification message has been sent.

Notification messages consist of an *error code* and an *error subcode* which further qualifies the main error. The format of notification messages is shown in Figure 376 on page 517.



Number of octets	
1	Error Code
1	Error Subcode
Variable	Data

Figure 376. BGP Notification Message

Error codes that are provided by BGP are as follows:

- Message Header Error
- Open Message Error
- Update Message Error
- Hold Timer Expired
- Finite State Machine Error
- Cease

A *data field* is included in the notification message to provide additional diagnostic information.

B.6.3 IBM 8210 and BGP Version 4

The IBM 8210 supports a full implementation of BGP Version 4 with *null authentication*. Only BPG Version 4 support is provided, and earlier BPG versions are not supported.

Configuration options exist to:

- Enable BGP and specify the local Autonomous System number
- Define BGP neighbors
- Define (Exclude) ASs from which no routing information will be accepted
- Define send, receive, and originate policies

  See B.6.1.1, “Originate, Send and Receive Policies” on page 509.
- Define aggregate routes

  The IBM 8210 IP router requires that aggregated routes (see Figure 375 on page 516) are preconfigured. When defining aggregated routes make sure that the individual routes that make up the aggregated route are not exported.

For details on configuring IBM 8210 BGP functions, see 7.4.2, “BGP Configuration on the IBM 8210” on page 191.

Appendix C. Special Notices

This publication is intended to help people involved in the design, construction, management, and operation of ATM networks using the IBM 8210 Nways MSS Server. The information in this publication is not intended as the specification of any hardware or software interface provided by the IBM 8210 Nways MSS Server. See the PUBLICATIONS section of the IBM Programming Announcement for the IBM 8210 Nways MSS Server. for more information about what publications are considered to be product documentation.

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 program that does not infringe any of IBM′s intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (″vendor″) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer′s ability to evaluate and integrate them into the customer′s operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX
IBM
OS/2
WebExplorer
DatagLANce
Nways
RS/6000
WIN-OS/2

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

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

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

Java and HotJava are trademarks of Sun Microsystems, Inc.

Netscape	Netscape Company

Other trademarks are trademarks of their respective companies.

Appendix D. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

D.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see “How To Get ITSO Redbooks” on page 523.

- *TCP/IP Tutorial and Technical Overview*, GG24-3376
- *IBM Multisegment LAN Design Guidelines*, GG24-3398
- *ATM Technical Overview*, SG24-4625
- *Campus ATM Design Guidelines*, SG24-5002
- *IBM 8260 As a Campus ATM Switch*, SG24-5003

D.2 Redbooks on CD-ROMs

Redbooks are also available on CD-ROMs. **Order a subscription** and receive updates 2-4 times a year at significant savings.

CD-ROM Title	Subscription Number	Collection Kit Number
System/390 Redbooks Collection	SBOF-7201	SK2T-2177
Networking and Systems Management Redbooks Collection	SBOF-7370	SK2T-6022
Transaction Processing and Data Management Redbook	SBOF-7240	SK2T-8038
AS/400 Redbooks Collection	SBOF-7270	SK2T-2849
RISC System/6000 Redbooks Collection (HTML, BkMgr)	SBOF-7230	SK2T-8040
RISC System/6000 Redbooks Collection (PostScript)	SBOF-7205	SK2T-8041
Application Development Redbooks Collection	SBOF-7290	SK2T-8037
Personal Systems Redbooks Collection	SBOF-7250	SK2T-8042

D.3 Other Publications

These publications are also relevant as further information sources:

- *8210 Nways MSS Server Setup and Problem Determination Guide*, GA27-4140
- *8210 Nways MSS Server Operations Reference Card*, GX27-4017
- *Nways MSS Server Configuration Guide*, SC30-3821
- *Nways MSS Server Introduction and Planning Guide*, GC30-3820
- *Nways MSS Server Command Line Interface: User's Guide*, SC30-3818
- *Nways MSS Server Command Line Interface: Protocol Configuration Guide*, SC30-3819
- *Event Logging System Messages Guide*, SC30-3682
- *Nways MSS Server Service Manual*, GY27-0354
- *MSS Server Module Setup Guide*, GA27-4141

How To Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at URL http://www.redbooks.ibm.com.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM
- **Tools disks**

  To get LIST3820s of redbooks, type one of the following commands:

  ```
  TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
  TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
  ```

  To get lists of redbooks:

  ```
  TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
  TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
  ```

  To register for information on workshops, residencies, and redbooks:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
  ```

  For a list of product area specialists in the ITSO:

  ```
  TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
  ```

- **Redbooks Home Page on the World Wide Web**

  http://w3.itso.ibm.com/redbooks

- **IBM Direct Publications Catalog on the World Wide Web**

  http://www.elink.ibmlink.ibm.com/pbl/pbl

  IBM employees may obtain LIST3820s of redbooks from this page.

- **REDBOOKS category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

  With an Internet E-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an E-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** (Do not send credit card information over the Internet) — send orders to:

	IBMMAIL	**Internet**
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications
Publications Customer Support
P.O. Box 29570
Raleigh, NC 27626-0570
USA

IBM Publications
144-4th Avenue, S.W.
Calgary, Alberta T2P 3N5
Canada

IBM Direct Services
Sortemosevej 21
DK-3450 Allerød
Denmark

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States)** or **(+1) 415 855 43 29 (Outside USA)** — ask for:

  Index # 4421 Abstracts of new redbooks
  Index # 4422 IBM redbooks
  Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com

- **On the World Wide Web**

Redbooks Home Page	http://www.redbooks.ibm.com
IBM Direct Publications Catalog	http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Internet Listserver**

  With an Internet E-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an E-mail note to announce@webster.ibmlink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

IBM Redbook Order Form

Please send me the following:

Title	**Order Number**	**Quantity**

- **Please put me on the mailing list for updated versions of the IBM Redbook Catalog.**

First name Last name

Company

Address

City Postal code Country

Telephone number Telefax number VAT number

- Invoice to customer number

- Credit card number

Credit card expiration date Card issued to Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

DO NOT SEND CREDIT CARD INFORMATION OVER THE INTERNET.

List of Abbreviations

AAL	ATM Adaptation Layer
ABR	Available Bit Rate
ACM	Address Complete Message
ACPSW	ATM Control Point and Switch
ADPCM	Adaptive Differential Pulse Code Modulation
AIM	ATM Inverse Multiplexer
AIR	Additive Increase Rate
AIS	Alarm Indication Signal
AIX	Advanced Interactive eXecutive
AMI	Alternate Mark Inversion
ANI	Automatic Number Identification
ANM	Answer Message
ANSI	American National Standards Institute
API	Application Programming Interface
APPN	Advanced Peer to Peer Network
ARE	All Routes Explorer
ARP	Address Resolution Protocol
ARQ	Automated Repeat reQuest
ASCII	American Standard Code for Information Interchange
ASIC	Application Specific Integrated Circuit
ASN	Abstract Syntax Notation
ASP	Abstract Service Primitive
ASRT	Adaptive Source Routing Transparent
ATD	Asynchronous Time Division
ATM	Asynchronous Transfer Mode
ATMARP	ATM Address Resolution Protocol
B-ICI	Broadband Inter Carrier Interface
B-ICI SAAL	B-ICI signaling ATM Adaptation Layer
B-ISDN	Broadband Integrated Services Digital Network
B-LLI	Broadband Low Layer Information
B-NT	Broadband Network Termination
B-TE	Broadband Terminal Equipment
BCBDS	Broadband Connectionless Data Bearer Service
BCD	Binary Coded Decimal
BCM	BroadCast Manager
BCOB	Broadband Class of Bearer
BECN	Backward Explicit Congestion Notification
BER	Bit Error Rate
BGP	Border Gateway Protocol
BISDN	Broadband - Integrated Services Digital Network
BISSI	Broadband Inter Switching System Interface
BN	Bridge Number
BOM	Beginning of Message
BOOTP	Bootstrap Protocol
BPDU	Bridge Protocol Data Unit
BRI	Basic Rate Interface
BSVC	Broadcast Switched Virtual Connections
BT	Burst Tolerance
BTAG	Begin Tag
BUS	Broadcast and Unknown Server
BW	Bandwidth
CA	Cell Arrival
CAC	Connection Admission Control
CBDS	Connectionless Broadband Data Service
CBR	Constant Bit Rate
CCITT	Consultative Committee on International Telephone & Telegrap
CCR	Current Cell Rate
CCS	Common Channel Signaling
CCSS7	Common Channel Signaling System 7
CDT	Cell Delay Tolerance
CDV	Cell Delay Variation

CDVT	Cell Delay Variation Tolerance
CEI	Connection Endpoint Identifier
CER	Cell Error Ratio
CES	Circuit Emulation Service
CI	Congestion Indicator
CIP	Carrier Identification Parameter
CIR	Committed Information Rate
CL	Connectionless
CLNAP	Connectionless Network Access Protocol
CLNP	Connectionless Network Protocol
CLNS	Connectionless Network Service
CLP	Cell Loss Priority
CLR	Cell Loss Ratio
CLS	Connectionless Server
CLSF	Connectionless Service Function
CMI	Coded Mark Inversion
CMIP	Common Management Interface Protocol
CMR	Cell Misinsertion Rate
CO	Connection Oriented
COM	Continuation of Message
COS	Class of Service
CP	Connection Processor
CP	Control Point
CPCS	Common Part Convergence Sublayer
CPE	Customer Premises Equipment
CPG	Call Progress Message
CPI	Common Part Indicator
CPN	Customer Premises Network
CPN	Calling Party Number
CPSW	Control Point Switch
CRC	Cyclic Redundance Check
CRCG	Common Routing Connection Group
CS	Convergence Sublayer
CS	Carrier Selection
CS1	Capability Set One
CS2	Capability Set Two
CSI	Convergence Sublayer Indication
CSPDN	Circuit Switched Public Data Network
CSR	Cell Missequenced Ratio
CSU	Channel Service Unit
CTD	Cell Transfer Delay
CTV	Cell Tolerance Variation
DA	Destination MAC address
DA	Destination Address
DCC	Data Country Code
DCE	Data Communication Equipment
DD	Depacketization Delay
DLC	Data Link Control
DES	Destination End System
DLCI	Data Link Connection Identifier
DOS	Disk Operating System
DQDB	Distributed Queue Dual Bus
DS	Distributed Single Layer Test Method
DS-0	Digital Signal, Level O
DS-1	Digital Signal, Level 1
DS-2	Digital Signal, Level 2
DS-3	Digital Signal, Level 3
DS3 PLCP	Physical Layer Convergence Protocol
DSE	Distributed Single Layer Embedded Test Method
DSID	Destination Signaling Identifier
DSS2	Setup Digital Subscriber Signaling #2
DSU	Data Service Unit
DTE	Data Terminal Equipment
DXI	Data Exchange Interface
EBCDIC	Extended Binary Coded Decimal Interchange Code
EFCI	Explicit Forward Congestion Indication
ELAN	Emulated Local Area Network
ELS	Event Logging System
EMI	Electromagnetic Interference
EOM	End of Message
ESI	End System Identifier

ETAG	End Tag
FC	(ATM) Forum Compliant
FCS	Fast Circuit Switching
FCS	Frame Check Sequence
FDDI	Fiber Distributed Data Interface
FEBE	Far End Block Error
FEC	Forwarding Engine
FEC	Forward Error Correction
FERF	Far End Receive Failure
FRS	Frame Relay Service
FTP	Foiled Twisted Pair
FUNI	Frame User Network Interface
GAP	Generic Address Parameter
GCID	Global Call Identifier
GFC	Generic Flow Control
HDB3	High Density Bipolar 3
HDLC	High Level Data Link Control
HEC	Header Error Control
HEC	Header Error Check
HEL	Header Extension Length
HLPI	Higher Layer Protocol Identifier
HOL	Head of Line
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IAA	Initial Address Acknowledgment
IAM	Initial Address Message
IAR	Initial Address Reject
IBM	International Business Machines Corporation
IBUS	Intelligent Broadcast and Unknown Server
IC	Initial Cell Rate
ICD	International Code Designator
ICMP	Internet Control Message Protocol
IDU	Interface Data Unit
IE	Information Element
IEC	Inter-exchange Carrier
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ILES	Inetlligent LAN Emulation Server
ILMI	Interim Link Management Interface
ILMI	Interim Local Management Interface
IOP	Interoperability
IP	Internet Protocol
IPng	Internet Protocol Next Generation
IPX	Novell Internetwork Packet Exchange
ISO	International Organization for Standardization
ITSO	International Technical Support Organization
ITU	International Telecommunications Union
ITU-T	International Telecommunications Union - Telecommunication
IUT	Implementation Under Test
IWF	Interworking Function
IWU	Interworking Unit
JPEG	Joint Photographic Experts Group
LAN	Local Area Network
LANE	Local Area Network Emulation
LAPD	Link Access Procedure D
LB	Leaky Bucket
LD	LAN Destination
LE	LAN Emulation
LE-ARP	LAN Emulation Address Resolution Protocol
LEC	LAN Emulation Client
LEC	Local Exchange Carrier
LECID	LAN Emulation Client Identifier
LECS	LAN Emulation Configuration Server
LED	Light Emitting Diode
LES	LAN Emulation Server
LIJP	Leaf Initiated Join Parameter
LIS	Logical IP Subnet
LIV	Link Integrity Verification
LLATMI	Lower Layer ATM Interface
LLC	Logical Link Control

LLC/SNAP	Logical Link Control/Subnetwork Access Protocol
LMI	Layer Management Interface
LNNI	LAN Emulation Network-to-Network Interface
LOC	Loss of Cell delineation
LOF	Loss of Frame
LOS	Loss of Signal
LSB	Least Significant Bit
LSR	Leaf Setup Request
LUNI	LAN Emulation User-to-Network Interface
MAC	Medium Access Control
MAN	Metropolitan Area Network
MBS	Maximum Burst Size
Mbps	Megabits Per Second
MB	Megabytes
MCR	Minimum Cell Rate
MCTD	Mean Cell Transfer Delay
MIB	Management Information Base
MID	Message Identifier
MIR	Maximum Information Rate
MMF	Multimode Fiberoptic cable
MPEG	Motion Picture Experts Group
MPOA	Multiprotocol over ATM
MRCS	Multi-rate Circuit Switching
MSAP	Management Service Access Point
MSB	Most Significant Bit
MSN	Monitoring Cell Sequence Number
MSS	Multiprotocol Switching Services
MSVC	Meta-signaling Virtual Channel
MT	Message Type
MTU	Message Transfer Unit
N-ISDN	Narrowband Integrated Services Digital Network
NBBS	Network BroadBand Services
NCE	Network Control Engine
NDIS	Network Driver Interface Specification
NETBIOS	Network Basic Input/Output System
NHRP	Next Hop Resolution Protocol
NMS	Network Management System
NNI	Network to Network Interface
NP	Network Performance
NPC	Network Parameter Control
NRB	Non Reserved Bandwidth
NRM	Network Resource Management
NSAP	Network Service Access Point
NSP	Network Service Provider
NSR	Non-Source Routed
NT	Network Termination
OAM	Operations, Administration and Maintenance
OC-n	Optical Carrier level n
ODI	Open Data-Link Interface
OSI	Open systems Interconnection
OSPF	Open Shortest Path First
OS/2	Operating System/2
OUI	Organization Unique Identifier
OUI	Organizational Unit Identifier
P-NNI	Private Network to Network Interface
PAD	Packet Assembler and Disassembler
PBX	Private Branch eXchange
PC	Priority Control
PC	Protocol Control
PCM	Pulse Code Modulation
PCMCIA	Personal Computer Memory Card International Association
PCO	Point of Control and Observation
PCR	Peak Cell Rate
PCR	Program Clock Reference
PCVS	Point to Point Switched Virtual Connections
PD	Packetization Delay
PDH	Plesiochronous Digital Hierarchy
PDU	Packet Data Unit
PDU	Protocol Data Unit
PHY	Physical Layer

PIN	Personal Identification Number
PL	Physical Layer
PLL	Phase Locked Loop
PLPC	Physical Layer Convergence Protocol
PNNI	Public Network-to-Network Interface
POH	Path Overhead
POI	Path Overhead Indicator
PT	Payload Type
PTI	Payload Type Identifier
PVC	Permanent Virtual Circuit
PVCC	Permanent Virtual Channel Connection
PVPC	Permanent Virtual Path Connection
QD	Queuing Delay
QOS/QoS	Quality of Service
RB	Reserved Bandwidth
RBOC	Regional Bell Operating Company
RC	Routing Control
RD	Route Descriptor
RDF	Rate Decrease Factor
RDI	Remote Defect Identification
REL	Release Message
RFC	Request For Comment
RFI	Radio Frequency Interference
RGB	Red Green Blue
RI	Routing Information
RIF	Routing Information Field
RII	Routing Information Indicator
RIP	Routing Information Protocol
RISC	Reduced Instruction Set Computing
RNR	Receive Not Ready
RSVP	Resource Reservation Protocol
RR	Receive Ready
RT	Routing Type
RTS	Residual Time Stamp
S-FTP	Screened Foiled Twisted Pair
SA	Source MAC address
SA	Source Address
SAAL	Signaling ATM Adaptation Layer
SAP	Service Access Point (IEEE)
SAP	Service Advertisment Protocol (Novell)
SAR	Segmentation and Reassembly
SCCP	Signaling Connection and Control Part
SCP	Service Control Point
SCR	Sustainable Cell Rate
SDH	Synchronous Digital Hierarchy
SDU	Service Data Unit
SE	Switching Element
SEAL	Simple and Efficient Adaptation Layer
SEL	Selector (byte)
SF	Switching Fabric
SIPP	SMDS Interface Protocol
SIR	Sustained Information Rate
SLIP	Serial Link Internet Protocol
SMC	Sleep Mode Connection
SMDS	Switched Multi-megabit Data Services
SMF	Single Mode Fiber
SN	Sequence Number
SNA	Systems Network Architecture
SNAP	Sub Network Access Protocol
SNMP	Simple Network Management Protocol
SOH	Section Overhead
SONET	Synchronous Optical Network
SPID	Service Protocol Identifier
SPTS	Single Program Transport Stream
SR	Source Routing (Bridging)
SRB	Source Routing Bridge
SRF	Specifically Routed Frame
SRM	Source Route Manager
SRT	Source Routing Transparent
SRTS	Synchronous Residual Time Stamp
SSCF	Service Specific Coordination Function

SSCOP	Service Specific Connection Oriented Protocol
SSCS	Service Specific Convergence Sublayer
SSI	Switch to Switch Interface
ST	Segment Type
STB	Spanning Tree Bridge
STE	Spanning Tree Explorer
STM	Synchronous Transfer Mode
STM1	Synchronous Transport Mode 1 -- 155mbits/sec
STP	Signaling Transfer Point
STP	Shielded Twisted Pair cable
STP	Spanning Tree Protocol
STS	Synchronous Time Stamps
STS-3c	Synchronous Transport System-Level 3 concatenated
SVC	Switched Virtual Circuit
SVCI	Switched Virtual Circuit Identifier
SVN	Switched Virtual Network
SVP	Switched Virtual Path
T1S1	ANSI T1 Subcommittee
TB	Transparent Bridging
TC	Transmission Convergence
TCP	Transmission Control Protocol
TCP	Test Coordination Procedure
TCP/IP	Transmission Control Program/Internet Protocol
TCS	Transmission Convergence Sublayer
TDJ	Transfer Delay Jitter
TDM	Time Division Multiplexing
TE	Terminal Equipment
TFTP	Trivial File Transfer Protocol
TLV	Type - Length - Value
TM	Traffic Management
TNS	Transit Network Selection
TPCC	Third Party Call Control
TS	Traffic Shaping
TS	Time Stamp
TS	Transport Stream
TS	Time Slot
TSAP	Transport Service Access Point
TTY	TeleTYpe
UBR	Unspecified Bit Rate
UDP	User Datagram Protocol
UNI	User Network Interface
UPC	Usage Parameter Control
UT	Upper Tester
UTOPIA	Universal Test & Operations PHY Interface for ATM
UTP	Unshielded Twisted Pair cable
VBR	Variable Bit Rate
VBR non-interactive	Variable Bit Rate non-interactiv
VC	Virtual Channel (Virtual Circuit)
VCC	Virtual Channel Connections
VCI	Virtual Circuit Identifier
VCI	Virtual Connection Identifier
VCI	Virtual Channel Identifier
VLAN	Virtual Local Area Network
VP	Virtual Path
VPC	Virtual Path Connection
VPCI	Virtual Path Connection Identifier
VPI	Virtual Path Identifier
WAN	Wide Area Network
WWW	World Wide Web

Index

Printed in U.S.A.