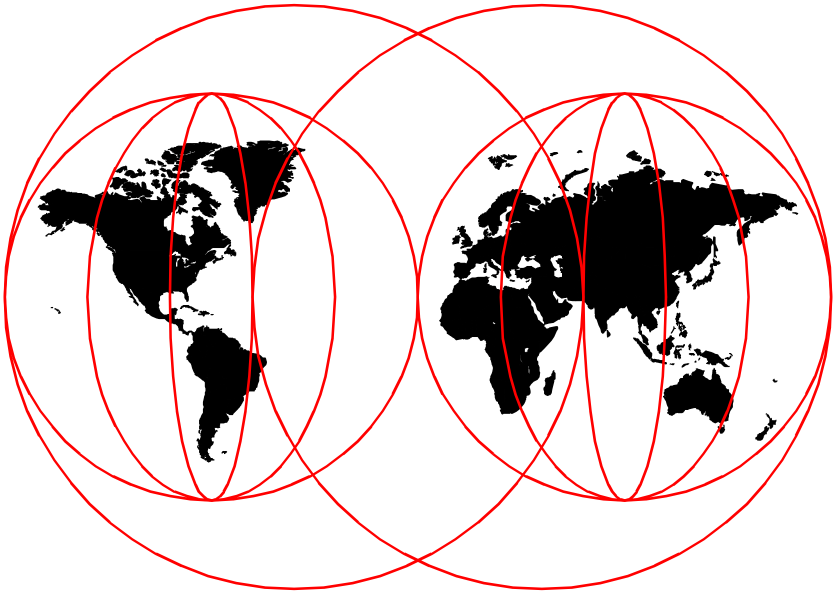
IBM Router Interoperability and Migration Examples

Jonathan Follows, Marcus Schmid, David Vieritz



International Technical Support Organization

www.ibm.com/redbooks

SG24-5865-00

SG24-5865-00

International Technical Support Organization

IBM Router Interoperability and Migration Examples

April 2000

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 167.

First Edition (April 2000)

This edition applies to Version 3 Release 4 of MRS, MAS and AIS software for the 2210 Nways Multiprotocol Router, 2216 Nways Multiaccess Connector and 2212 Access Utility respectively and to Cisco router code (such as IOS 12.1).

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 written from the viewpoint of an organization with an installed IBM router network that needs to grow the network’s size or add new functions (such as the transport of voice traffic). We are also assuming that this will be accomplished by adding router equipment manufactured by other vendors. We assume that the reader is already familiar with the models, features and configuration of IBM routers and we consider the issues faced by the organization when selecting and integrating other router equipment into the existing network.

In reality, this book specifically discusses adding Cisco routers to an existing IBM router network. We did not want to exclude other router manufacturers, but the realities of existing market share, the time available to the authors of this book, and the growing IBM/Cisco Alliance led us to give particular emphasis to familiarizing the reader with the range of Cisco router products and giving the reader a broad understanding of the models and features which complement existing IBM router network configurations.

This redbook helps you to install, tailor and configure Cisco routers to add to an existing network of IBM routers. It is assumed that the reader is unacquainted with Cisco products so the Cisco router configuration procedure is described in more detail than the corresponding IBM router configuration. However, this redbook does not aim to replace existing Cisco documentation and also does highlight specific IBM configuration requirements when appropriate.

Extensive examples are provided to assist with configuration of:

- Dynamic Routing Protocols including EIGRP, OSPF and RIP.
- Data Link Switching (DLSw) to transport SNA traffic through a mixed network of IBM and Cisco routers.
- APPN functions using the new “SNA Switching” subsystem now available with Cisco IOS software
- Voice transport through frame relay and IP networks of IBM and Cisco routers.

Discussion of dial-up connectivity and Novell IPX is also included.

This redbook will help you design a network which makes best use of the features of both IBM and Cisco routers to allow interoperability and migration between them to be as simple as possible.

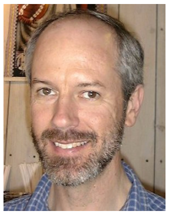
This edition is not directly related to a specific code release, but most use is made of the latest releases of IBM router code (Version 3 Release 4 of MRS, MAS and AIS for the 2210, 2216 and 2212 respectively) and of Cisco router code (such as IOS 12.1). Where specific code levels are required for particular functions, these code level requirements are made clear in the text of the book.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Jonathan Follows is an IBM-certified networking specialist at the International Technical Support Organization, Raleigh. He writes redbooks on all areas of IBM networking hardware and software, most recently on policy-based networking. Before joining the ITSO in 1998, he worked as a technical specialist providing sales and marketing support in the United Kingdom and has 15 years' experience in all types of networking. Jonathan read Mathematics at Oxford University, England, and holds a degree in Computing Science from London University, England.

Marcus Schmid is a Networking Consultant with IBM Unternehmensberatung GmbH (IBM UBG). Before joining IBM UBG he was a scientist at the IBM European Networking Center in Heidelberg, Germany. He has four years of experience in the networking environment. He holds an MS degree in electrical engineering from the University of Stuttgart and an MS degree in telecommunications from the Ecole Nationale Superieure des Telecommunications in Paris, France.



David Vieritz is a Networking Specialist in IBM Australia's Systems Management and Network Services division. His responsibilities include network architecture and design, migration, implementation and ongoing support of IBM's networking customers. He has eight years of networking experience with IBM including TCP/IP and SNA network design as well as switched LAN and ATM implementation experience. He holds a degree in Electrical Engineering from the University of Queensland.

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

Gail Christensen, Harri Levanen, Shawn Walsh, Tatsuhiko Kakimoto, Bob Haimowitz
International Technical Support Organization, Raleigh Center

Flemming Schultze, Erik Bruun
IBM Denmark

Jerzy Buczak
Formerly ITSO, Raleigh

John Zeiger, Gary Coil, Andy Vrabel, Gregg Miller
IBM Networking Hardware Division

Toni Sweetland
Formerly IBM NHD

Matthias Binzer, Mike Blair, Ron Howell, Dave Kaminski, Steve Koretsky
Cisco Systems

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Please send us your comments about this or other Redbooks in one of the following ways:

- Fax the evaluation form found in "IBM Redbooks review" on page 179 to the fax number shown on the form.
- Use the online evaluation form found at http://www.ibm.com/redbooks
- Send your comments in an Internet note to redbook@us.ibm.com

Chapter 1. Migration strategies

This book is written for a specific audience: a technical audience of people who are somewhat familiar with IBM routers but not necessarily familiar with Cisco routers. It should be noted that this audience would have included the authors of this book before the writing started; this book is in part a document of our experiences setting up mixed IBM and Cisco router networks. The book is written for people involved with organizations that have existing networks based on IBM routers in the 2210, 2212 and 2216 router families.

Migration to routing platforms manufactured by other vendors and the integration of these routers into the existing IBM router network raise issues of compatibility and interoperability for a wide range of network protocols and functions. In this book we focus on the integration of Cisco routers into the IBM router network and supply the reader with configuration examples designed to provide compatibility between all routers in the network. Many configurations that include mixed networks of IBM and Cisco routers are documented and interoperability issues identified.

We assume that the reader is not familiar with Cisco router hardware and software, so we provide more detail in the Cisco configuration examples than the matching IBM router configurations. Cisco manufactures a wide range of hardware routing platforms and offers a wide range of software feature sets for these routers. In this first chapter we provide an overview of some of these Cisco router hardware and software offerings and attempt to match them with comparable IBM router platforms that the reader will be familiar with. We do not cover the entire Cisco range in this chapter, but rather select the most common Cisco routers that existing IBM customers are likely to add to IBM router networks to provide a broad understanding of the customer choices.

1.1 Comparison of IBM and Cisco router hardware platforms

This section reviews the hardware and software features available on existing IBM routers and comparisons with comparable Cisco router products. Its purpose is to assist the reader who is familiar with IBM routers to configure and select appropriate comparable router products to add to an IBM router network.

Three of the most significant points of difference are:

1. Serial ports on Cisco routers are not all universally configurable via software for synchronous or asynchronous operation as is the case with IBM 2210s and 2212s. Any standard serial port on an IBM 2210 or 2212

can be configured via software for synchronous operation (frame relay or digital leased line services) at up to 2 Mbps (or greater) or for asynchronous operation to support dial-up router-to-router links or remote LAN access ports. Cisco modular access routers support two basic types of WAN Interface Cards (WICs):

   a. WIC-1T or WIC-2T, which support synchronous and asynchronous operation on all Cisco 1600 and 1700 series routers, but synchronous operation only on Cisco 2600 and 3600 series routers. WIC-1T and WIC-2T interfaces support synchronous operation at up to 2 Mbps.

   b. WIC-2A/S, which supports both lower-speed synchronous operation (up to 128 kbps) and asynchronous operation on the complete range of Cisco 1600, 1700, 2600 and 3600 routers.

2. The standard AUX port on Cisco 1700, 2600 and 3600 routers is configurable as an asynchronous serial port and can be attached to an analog modem to support general dial-up access to the router network. This can be particularly useful as a built-in port to support an automatic dial backup link in the event that a primary high-speed link fails. IBM routers, such as the 2210 and 2212, do not have an equivalent port. The console ports on IBM 2210 and 2212 routers are designed to provide only console access to the router configuration and monitoring system and can not be used for general in-band data communications. Note, however, that the smaller Cisco 1600 router series does not include an AUX port.

3. Software Features - All software for the IBM 2210 and 2212 routers is included in the price of the router and available for download from the IBM Web site[1]. Cisco routers have a wide range of different software options, which are described in more detail in 1.2, “Cisco router software overview” on page 8. Many of these are chargeable software features. It should be noted, however, that the IP feature set that includes a wide range of IP routing functions is included with each router and is a no charge item. Many organizations adding Cisco 2600 or 3600 series routers to an IBM router network are likely to require the Cisco *IP Plus* software image as it includes IBM features such as DLSw (but not APPN) and it also includes Voice over IP and voice over frame relay support.

1.1.1 Small office routers

The IBM 2210 Model 1S4/1U4 and 1S8/1U8 are the most common IBM routers in this category. They are fixed-configuration small office routers that offer one Ethernet interface, one ISDN basic rate interface and one serial interface as indicated in Table 1 on page 3. The 2210 Models 1S4/1U4 have a fixed 2 MB of flash memory and 4 MB of RAM and are generally suitable for

[1] With the exception of a few separately priced features such as Network Dispatcher or TN3270 Server.

running basic IP and bridging configurations. The Models 1S8/1U8 have twice the flash and RAM and are suitable for running more sophisticated configurations including Data Link Switching, multiprotocol routing and bridging as well as *Dials* remote LAN access functions.

Table 1. 2210-1Sx configurations

Model	**LAN**	**No.of WANS**	**ISDN BRI**	**Flash/DRAM**
2210-1S4/1U4	1 Ethernet	1 Serial	1 BRI	2/4 MB
2210-1S8/1U8	1 Ethernet	1 Serial	1 BRI	4/8 MB

The 2210-1S4/1U4 and 2210-1S8/1U8 do not support APPN.

Current model Cisco routers that are designed for use in the small office environment include:

- Cisco 1600
- Cisco 1720
- Cisco 1750

The Cisco 1600 is the closest equivalent to the small 2210. It is available in a range of fixed configurations. The Cisco 1600R series is the latest version of the 1600 range. The Cisco 1600R series operate in a similar fashion to the 2210 in that they store their software image on flash memory and operate from RAM. The original 1600 range operated from flash memory. The “R” series are lower priced and offer greater performance than the original models. IBM 2210 routers have always stored their software images on flash memory and operated from RAM.

A summary of the interface configurations of the Cisco 1600R range are shown in Table 2.

Table 2. Cisco 1600R router interfaces

	1601R	1602R	1603R	1604R	1605R
Fixed LAN	Ethernet 10Base-T (RJ45) and AUI (DB-15)				
Fixed WAN Interface	Sync/Async	56K	ISDN BRI S/T	ISDN BRI U	Ethernet 10Base-T
Optional Second WAN	Yes	Yes	Yes	Yes	Yes

The WAN option slot on the Cisco 1600 range supports a variety of WAN interface cards, which can allow the Cisco 1600 to be configured to match the interface specification of the IBM 2210 1Sx/1Ux routers or in a variety of other configurations. One useful example is the 1601R configured with a WIC-1T providing a second sync/async interface. This configuration would support an Ethernet branch office LAN with a primary WAN connected to a frame relay service and the second WAN interface configured to provide dial backup support via an analog modem. This option is not available on the IBM 2210-1Sx/1Ux routers. The 1600 series WIC modules are also used in all Cisco routers up to the 3600 range, and in certain modules of the larger 7200 and 7500 series. Although a number of two-port WIC modules exist for Cisco routers, these cannot be used in the Cisco 1600 router as its design allows for only one additional WAN interface to be installed in the WAN option slot.

The Cisco 1600 range also offers memory upgrade options whereas the IBM 2210-1Sx/1Ux provide fixed memory configurations to minimize the cost.

The Cisco 1720 is a modular router with a fixed 10/100 Ethernet port but no fixed WAN interfaces. It has two WAN module slots, supporting two WAN modules for increased interface flexibility.

Supported WAN modules include:

- 1 port serial (WIC-1T)
- 2 port serial (WIC-2T)
- 2 port sync/Async (WIC-2A/S)
- 1 port ISDN BRI
- 1 port 64/56 kbps CSU/DSU
- 1 port T1/Ft1

It also has a built-in AUX port that supports asynchronous modem access to the Cisco 1720 and can be used for in-band communications. This can be very useful as a built-in dial-backup interface.

The Cisco 1720 does not have same WAN interface limit (a maximum of two WAN interfaces) of the Cisco 1600 series; the Cisco 1720 supports up to four synchronous or asynchronous interfaces via the dual-port WIC modules, plus the AUX async port giving it interface configurations more closely matched to the IBM 2210 Model 24T or 24E routers.

The Cisco 1720 also has a much more powerful processor than the Cisco 1600 range and is suitable for higher load encryption applications to support high-volume VPNs.

The Cisco 1750 is the next step up from the Cisco 1720. It offers similar features to the Cisco 1720 and adds support for voice interfaces. It has two WAN module slots supporting the same range of WAN modules as the Cisco 1720, but these slots also support voice interface cards (VICs). A third slot is added specifically to support voice interface cards. Combinations of voice and WAN interface modules may be installed with support for a maximum of four voice interfaces or four WAN interfaces on the router.

Supported voice interface cards (VICs) provide up to four analog voice ports (FXS, FXO or E&M) to support small office key systems, PABXs, or telephone handsets.

The Cisco 1750 router also includes the built-in AUX port that can be useful for dial-in or dial-backup communications.

1.1.2 Medium office and modular access routers

This section reviews the configuration options relevant to customers who plan to add modular Cisco access routers to an existing IBM router network consisting of the following IBM router models:

- 2210-12E/12T
- 2210-128/129
- 2210-24T/24E/24M
- 2212-10H/10F
- 2212-40H/40F

These IBM routers provide a range of LAN and serial interfaces and routing performance ranging from approximately 3000 to 30000 packets per second.

The Cisco 2600 range of routers most closely match the configurations available on these IBM routers. The 2600 range consists of the four base chasses shown in Table 3.

Table 3. Cisco 2600 router models

Model	Ethernet	10/100 Ethernet	Token-ring	CPU
Cisco 2610	1			40 MHz Motorola MPC860
Cisco 2611	2			40 MHz Motorola MPC860
Cisco 2612	1		1	40 MHz Motorola MPC860
Cisco 2613			1	40 MHz Motorola MPC860
Cisco 2620		1		50 MHz Motorola MPC860
Cisco 2621		2		50 MHz Motorola MPC860

Each of these 2600 routers has two small WIC slots that support a range of single and two port serial data interfaces, such as the WIC-2T and the

WIC-2A/S. Each Cisco 2600 also has one larger network module slot and an advanced integration module slot.

The network module slot supports a wide range of network modules that include:

- 1 and 2 port T1/E1 or ISDN PRI modules
- 4 and 8 port ISDN BRI modules
- 8 and 16 analog modems
- 16 and 32 port high density asynchronous port modules
- 1 or 2 slot voice/fax network modules (4 or 8 voice channels)
- 24 or 48 channel T1 high density voice/fax network modules

These are the same network modules used in the Cisco 3600 range. Many of these network modules are really a base adapter that have slots to support smaller interface cards such as additional WICs or the Voice Interface Cards (VICs).

The advanced integration module is provided to specifically support a data compression processor that offloads the main CPU from data compression tasks.

1.1.3 High-end modular access routers

This section gives an introduction to the Cisco modular router models that can be most easily compared to the following IBM router models:

- 2212-15H/15F
- 2212-45H/45F

These IBM routers provide a range of LAN and serial interfaces and routing performance of > 100,000 packets per second.

The Cisco 3600 range of routers most closely match the configurations available on these IBM routers. The 3600 range consists of two base models:

- Cisco 3620 has two network module slots
- Cisco 3640 has four network module slots

Unlike the Cisco 2600 range, the 3600 routers do not have any fixed LAN interfaces. They utilize the extensive range of network modules to provide all of their LAN and WAN or voice interfaces. These are the same network modules as used by the Cisco 2600 range. Many of the network modules have slots to house WAN or voice interface cards (WICs or VICs) to provide the actual physical interfaces. The Cisco 3640 has a 100-MHz IDT R4700

RISC processor and supports 128 MB of RAM, whereas the 3620 has an 80-MHz processor and supports 64 MB of RAM.

Both Cisco 3600 router models have the standard AUX port that support in-band asynchronous communication at up to 115 kbps.

1.1.4 High-performance and channel-attached routers

The IBM 2216 Model 400 and Network Utility are generally used in environments that require S/390 channel attachment and as central office “catcher” routers for a network of 2210s and 2212s. The closest equivalent Cisco routers are the Cisco 7200 and 7500 ranges of routers.

1.1.4.1 Cisco 7200

The Cisco 7200 routers have four base components, the chassis, a Network Processing Engine (NPE) that is the CPU, an I/O controller, and port adapters.

The 7200 range consists of two base chassis configurations:

Cisco 7204 - four slot chassis

Cisco 7206 - six slot chassis

These high-end routers utilize a different adapter type to the network modules and WICs of the 2600 and 3600 range. The Cisco 7200 routers use port adapters that provide a wide range of high-density interfaces for WAN, LAN, ISDN, ATM and digital voice adapters. These are the same adapters that are used in the Versatile Interface Processor (VIP) of the larger 7500 range and provide commonality of adapters across these two ranges.

The CPU of the Cisco 7200 router is called a Network Processing Engine (NPE). There are six of these available, each with different amounts of RAM, cache and clock speeds.

The I/O controller is a mandatory part of a Cisco 7200 order. An I/O controller with an integrated 10/100 Ethernet port is also available, which is a lower cost option than adding a Fast Ethernet port adapter to the configuration. A network configuration that may have required a Cisco 7206 may be able to be serviced by a Cisco 7204 if the I/O Controller with integrated Ethernet port is utilized.

A number of Cisco 7200 packaged configurations are also available, including one with a S/390 channel adapter at a reduced price compared to the individual components. ESCON and parallel channel adapters in the Cisco 7200 range are called channel port adapters and each provides a single channel interface. A wide range of SNA and TCP/IP host gateway functions, approximately equivalent to those of the IBM 2216, are available on the Cisco 7200 with channel port adapter.

1.1.4.2 Cisco 7500

The Cisco 7500 range comprises the 7505, 7507 and 7513 Models. The main processing engine is called the Route Switch Processor (RSP), of which there are three models: RSP1, RSP2 and RSP4. All three models of RSP are externally-clocked at 100 MHz, but the RSP4 has an internal clock speed of 200 MHz. The 7505 has a single internal bus (CyBus, rated at 1.067 gigabits per second) and has slots for up to four interface cards; the RSP occupies the fifth slot and can either be an RSP1 or an RSP4. The 7507 has two independent CyBuses and two RSPs (either RSP2 or RSP4) can be installed; the second RSP can take over from the first in case of failure of the primary RSP. Up to five interface cards can be installed in the 7507. The 7513 is a larger version of the 7507 with slots for up to 11 interface cards. In all cases, the same IP and VIP cards can be used across the 7500 range and also across the 7200 range.

The 7513 can weigh up to 160 pounds when fully configured, requires a 20A power supply and consumes up to 1200W of power. It provides up to 88 ports.

1.2 Cisco router software overview

Cisco's router operating system, IOS (Internetworking Operating System), is available in a wide range of packaged forms. The packages vary depending on the Cisco router hardware model involved. The following sections review the basic software packages available for different Cisco router models with regard to the features that are likely to be required by organizations with existing IBM router networks. In all cases we only consider the packaging available with IOS 12.0.

1.2.1 Cisco 1600 and Cisco 1700 router series software

The basic software packages available for the Cisco 1600 series are:

- IP
- IP Plus
- IP/IPX
- IP/IPX/AT/IBM
- IP/IPX/AT/IBM Plus

Additional feature sets that add encryption and firewall functions are available. The “plus” feature set adds more advanced functions to the basic protocol set. Table 4 shows a summary of the functions that are included in each of the basic software packages for the Cisco 1600 Series.

Packaging on the Cisco 2600/3600 ranges changed in IOS Release 12 to include some IBM functions (such as DLSw) in the “IP Plus” package; the Cisco 1600 range retains the older packaging of IBM features in a separate IP/IPX/AT/IBM feature set, rather than including it in the “IP Plus” feature set.

Table 4. Commonly required IBM features and corresponding IOS packaging for Cisco 1600

Feature	**IP**	**IP Plus**	**IP/IPX**	**IP/IPX/ AT/IBM**	**IP/IPX/ AT/IBM Plus**
IP Routing	Y	Y	Y	Y	Y
RIP/OSPF/ EIGRP	Y	Y	Y	Y	Y
DLSw	N	N	N	Y	Y
NAT	N	Y	N	Y	Y
L2TP	N	Y	N	N	Y
IPX	N	N	Y	Y	Y

Note that the base IP feature set is available at no charge. Charges apply to upgrade to any of the other feature sets. APPN is not available within the Cisco 1600 or 1700 series routers, just as it is not available on the low-end IBM 2210-1S4/1S8or 1U4/1U8 routers.

The Cisco 1700 routers use similar packaging to the Cisco 1600 routers in that they require the IP/IPX/AT/IBM feature set to provide DLSw support. They also have a range of additional packages to provide voice support for the Cisco 1750. Again the base “data only” IP feature set is available at no charge. Upgrades to more advanced feature sets, including voice support are available at a charge.

1.2.2 Cisco 2600 and Cisco 3600 router series software

The latest IOS Release 12.0 software packaging available for the Cisco 2600 and Cisco 3600 routers includes the following basic packages:

- IP
- IP Plus
- IP/IPX/AppleTalk/DEC (sometimes called Desktop)

- IP/IPX/AppleTalk/DEC Plus
- Enterprise Plus
- Enterprise SNASw Plus

The *plus* option adds more advanced functions to the basic protocol features, and the Enterprise and Enterprise/SNASw versions of IOS are now only available with the Plus feature sets. A summary of functions that are likely to be required by organizations using IBM routers and the corresponding Cisco IOS software packages are shown in Table 5.

Table 5. Commonly required IBM features and corresponding IOS packaging

Feature	**IP**	**IP Plus**	**IP/IPX/ AT/DEC**	**IP/IPX/ AT/DEC Plus**	**Enterprise Plus**	**Enterprise SNASw Plus**
IP Routing	Y	Y	Y	Y	Y	Y
RIP/OSPF/EIGRP	Y	Y	Y	Y	Y	Y
DLSw	N	Y	N	Y	Y	Y
NAT	N	Y	N	Y	Y	Y
L2TP	N	Y	N	Y	Y	Y
VoIP VoFR	N	Y	N	Y	Y	Y
IPX	N	N	Y	Y	Y	Y
MPLS	N	N	N	N	Y	Y
APPN	N	N	N	N	N	Y

Additional encryption options are available for each of the packages shown in Table 5. The basic IP feature set is available for no charge with a Cisco router. Upgrades to more advanced feature sets are available at a charge.

Note that support for DLSw and for voice functions is added with the Plus feature and is not included in the basic IP feature set. It is likely that most organizations using IBM routers in a non-APPN network would require the basic IP package or the IP Plus package, which adds a number of IBM network features such as DLSw.

Additional special feature packages are available for such functions as firewall and IP switching.

1.2.3 Cisco 7200 and Cisco 7500 series router software

The latest IOS Release 12.0 software packaging available for the Cisco 7200 and 7500 routers includes the following basic packages:

- IP Standard Feature Set
- Enterprise Standard Feature Set
- Enterprise/SNASw Feature Set
- Desktop/IBM Standard Feature Set

Table 6. Commonly required IBM features and IOS packaging for Cisco 7200/7500 routers

Feature	IP	Enterprise	Enterprise SNASw	Desktop IBM
IP Routing	Y	Y	Y	Y
RIP/OSPF/ EIGRP	Y	Y	Y	Y
DLSw	Y	Y	Y	Y
NAT	Y	Y	Y	Y
L2TP	Y	Y	Y	Y
VoIP VoFR	Y	Y	Y	Y
IPX				
MPLS	N	Y	Y	N
APPN	N	N	Y	N

Additional encryption options are available for each of the packages shown in Table 6. The basic IP feature set is available at no charge with a Cisco router. Upgrades to more advanced feature sets are available at a charge.

Note that for the Cisco 7200 and 7500 Series routers basic IBM functionality is included in the base IP feature, including DLSw, so it is likely that many IBM customers may not require an additional software feature set. The basic IP set would satisfy many requirements. One significant item to note, however, is that to support frame relay protocols on Cisco 7200 or Cisco 7500 serial ports requires a Netflow/WAN Packet Protocol software license, which is a chargeable item. This is not required on the Cisco 1600, 1700, 2600 or 3600 router ranges for their frame relay support.

Also note that the APPN function provided by the Enterprise/SNASw feature set no longer provides full APPN function; instead the SNASw feature implements the Branch Extender (BrNN) function. This is described in more detail in Chapter 4, “APPN interoperability and migration” on page 97.

Chapter 2. Dynamic routing protocols

In this chapter we explore basic interoperability issues that arise as we incrementally add Cisco routers to an existing IP network consisting of IBM routers. We will investigate the following:

- Establishing WAN connectivity using frame relay or PPP between an existing IBM router network and a newly added Cisco router.
- Configuring dynamic routing protocols (RIP and OSPF) on Cisco routers to make them interoperate with existing IBM routers.
- Integrating complete systems using the Cisco proprietary routing protocol EIGRP into existing IBM OSPF networks.

A basic knowledge of the dynamic routing protocols RIP and OSPF is assumed.

2.1 Overview of the basic IBM network

Our starting point for the interoperability and migration examples is the router network depicted in Figure 1 on page 15. This network consists exclusively of IBM routers and has a star-like topology with a central IBM2216. Directly attached to the central IBM2216 are six branch routers (Berlin-IBM2210, Bonn-IBM2210, Boston-IBM2210, Melbourne-IBM2210, Munich-IBM2212, and Sydney-IBM2210). Behind Boston-IBM2210 router there are two cascaded branch routers (Chicago-IBM2212 and LA-IBM2210).

What we have tried to do with this network is to set up a mix of LAN and WAN networking so that although the topology of this network represents a typical customer environment, the mixture of employed WAN connection types and routing protocols does not, but serves to demonstrate a mix of possible networking environments in a single network:

- The branch routers Melbourne-IBM2210 and Sydney-IBM2210 are connected to the central site by means of a full mesh of frame relay permanent virtual circuits (PVCs). From the point of view of the OSPF routing protocol, these routers build a *Non-Broadcast Multiple Access (NBMA)* network. This NBMA network is contained in the IP subnet 10.1.4.0/24[1]. The PVCs between these three routers have a committed information rate (CIR) of 16 kbps.
- One PVC with a CIR of 16 kbps connects the branch routers Berlin-IBM2210, Bonn-IBM2210 and Munich-IBM2210 to the central

[1] The notation a.b.c.d/m for IP subnets is a short form for the subnet a.b.c.d with a subnet mask containing only “1” in the m most significant bits.

IBM2216. In terms of the OSPF routing protocol they comprise the *Point-to-Multipoint* network 10.1.7.0/24.

- The branch router Boston-IBM2210 is connected to the central IBM2216 with a 64 kbps PPP line. Two other branches– LA-IBM2210 and Chicago-IBM2212 – are linked to Boston with 64 kbps PPP lines.

NBMA and P2MP

A technical reminder:

Non-Broadcast Multiple Access (NBMA) is only used in frame relay (and similar non-broadcast) networks when the network is fully meshed, or in other words where there is a direct connection between each and every router connecting to the same NBMA network. It is not usually cost-effective to implement a fully meshed public frame relay network; on the other hand, if a fully meshed network is in place, then it is more efficient to implement OSPF as an NBMA configuration.

Point-to-Multipoint (P2MP) is the default configuration type for OSPF over frame relay using IBM routers. Even a fully meshed network can be configured as P2MP, but this simply means that the benefit of the full mesh is wasted because all traffic between pairs of branch routers will be required to pass through the central router anyway.

An NBMA implementation requires more configuration details to be provided than for a P2MP implementation, because the specific role of each router in the network needs to be predefined, and one or more routers must be defined as a Designated Router (DR). Fortunately much of this extra information need only be configured on the “central” router, such as the IBM 2216 in our example.

P2MP networks still require some configuration information on each router for the router’s partner across the network, but it is no longer necessary to define OSPF DRs because they do not exist in the P2MP environment.

As depicted in Figure 1 on page 15, the network is split into two autonomous systems. The routers Boston-IBM2210, Chicago-IBM2212, and LA-IBM2210 comprise an autonomous system in which the routing protocol RIPv2 is used; the remaining routers belong to the second autonomous system, in which the routing information is distributed with the OSPF protocol. The central IBM2216 redistributes the routes between both autonomous systems and as an area border router links the OSPF areas 0.0.0.1 and 0.0.0.2 to the OSPF backbone area.

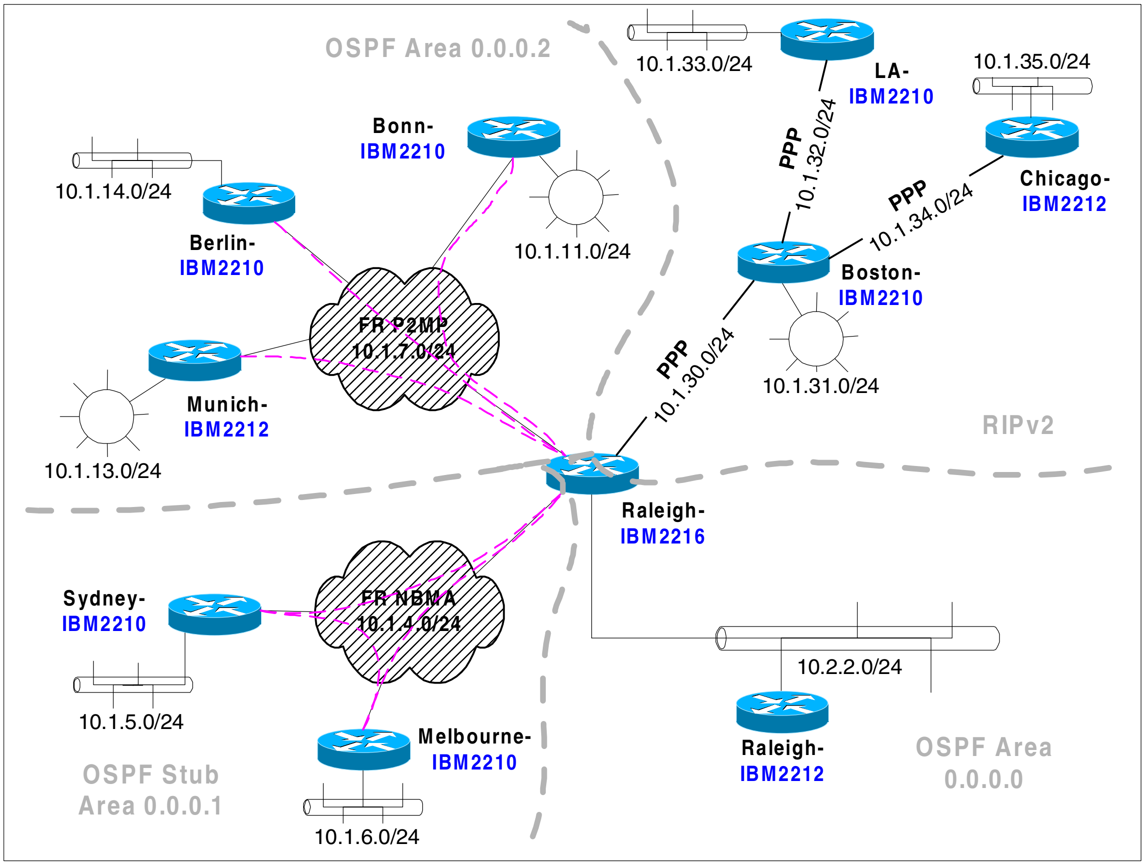


Figure 1. Overview of the basic IBM network

Table 7 lists the model types, software releases and internal addresses of all the routers of our basic IBM network. Note that the least significant octet of the IP address of any router in the network always matches the least significant octet of its internal address.

Table 7. Employed routers: model type, internal address, and software release

Router name	Internal address	Model	Software release
Berlin-IBM2210	10.1.255.1	2210-24M	MRS V3.3 (PTF NP01069)
Bonn-IBM2210	10.1.255.3	2210-24M	MRS V3.3 (PTF NP01069)
Boston-IBM2210	10.1.255.30	2210-24T	MRS V3.3 (PTF NP01069)

Router name	Internal address	Model	Software release
Chicago-IBM2212	10.1.255.32	2212 (45-2U)	AIS V3.3 (PTF NP01075)
LA-IBM2210	10.1.255.31	2210-12T	MRS V3.3 (PTF NP01069)
Melbourne-IBM2210	10.1.255.11	2210-12E	MRS V3.4
Munich-IBM2212	10.1.255.2	2212 (40-2U)	AIS V3.3 (PTF NP01075)
Raleigh-IBM2212	10.1.255.60	2212 (10-1U)	AIS V3.3 (PTF NP01075)
Raleigh-IBM2216	10.1.255.20	2216-400	MAS V3.4
Sydney-IBM2210	10.1.255.10	2210-24E	MRS V3.3 (PTF NP01069)

2.2 Adding a Cisco router to a RIP network

We start out by adding a Cisco branch router to an existing IBM router network using RIPv2. To show how the new Cisco router must be configured we replace Chicago-IBM2212 by a Cisco 2621 router. The configuration of Chicago-IBM2212 that has to be reproduced is as follows:

- Two interfaces of the IBM2212 are active: A Fast Ethernet interface with the IP address 10.1.35.32/24 and a serial interface using PPP encapsulation with the IP address 10.1.34.32/24.
- TCP header compression[2] (RFC1144) is enabled on the PPP link.
- RIPv2 is enabled on both interfaces and sends and receives net, subnet and host routes.
- The internal IP address of Chicago-IBM2212 is 10.1.255.32/32.

If we replace Chicago-IBM2212 with Chicago-Cisco2621, an analogous configuration looks like the one shown in Figure 2 on page 17. We describe this configuration dialog here in more detail than in the remainder of the book in order to illustrate the basic Cisco configuration process.

[2] Also called Van Jacobson compression, which can effectively reduce 20 bytes of IP header and 20 bytes of TCP header to 5 bytes or less. It has no effect on UDP packets, should not be confused with “RTP header compression”, and should always be enabled for low-speed PPP links. See Chapter 3, “Voice over IP and voice over frame relay” on page 41 for more information on RTP header compression.

The initial configuration dialog of the Cisco router requires specification of a password to enter the router (by a telnet or console session) and the `enable` password, which grants access to the privileged command mode. The configuration dialog in Figure 2 begins by entering the privileged command mode of the router, which is accomplished by typing in the command `enable` followed by the password you entered in the initial configuration dialog. Then you switch to the configuration mode of the router by typing `configure`. If you have not assigned a host name to the router in the initial configuration dialog, you can do this now - as we did - using the `hostname` command.

```
Router>enable
Password:
Router#configure
Configuring from terminal, memory, or network [terminal]? terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#hostname Chicago-Cisco2621
Router(config)#!
Chicago-Cisco2621(config)#interface FastEthernet0/0    1
Chicago-Cisco2621(config-if)#speed 100
Chicago-Cisco2621(config-if)#ip address 10.1.35.32 255.255.255.0
Chicago-Cisco2621(config-if)#no shutdown
Chicago-Cisco2621(config-if)#exit
Chicago-Cisco2621(config-if)#!
Chicago-Cisco2621(config)#interface Serial0/0    2
Chicago-Cisco2621(config-if)#encapsulation PPP
Chicago-Cisco2621(config-if)#ip address 10.1.34.32 255.255.255.0
Chicago-Cisco2621(config-if)#ip tcp header-compression
Chicago-Cisco2621(config-if)#no shutdown
Chicago-Cisco2621(config-if)#exit
Chicago-Cisco2621(config-if)#!
Chicago-Cisco2621(config)#interface Loopback0    3
Chicago-Cisco2621(config-if)#ip address 10.1.255.32 255.255.255.255
Chicago-Cisco2621(config-if)#exit
Chicago-Cisco2621(config)#!
Chicago-Cisco2621(config)#ip routing    4
Chicago-Cisco2621(config)#!
Chicago-Cisco2621(config)#router rip    5
Chicago-Cisco2621(config-rou)#version 2
Chicago-Cisco2621(config-rou)#network 10.0.0.0
Chicago-Cisco2621(config-rou)#exit
Chicago-Cisco2621(config)#end
Chicago-Cisco2621#
```

Figure 2. Configuration of Chicago-Cisco2621

To imitate the interface configuration of the IBM router, we first configure a Fast Ethernet interface. Therefore we enter the subconfiguration mode of the first Fast Ethernet interface by typing the command `interface FastEthernet0/0` 1. Note that the prompt changes to `Chicago-Cisco2621(config-if)`. In this mode we enter the speed and the IP address of the Fast Ethernet interface with the respective commands. The `no shutdown` command activates the interface. Then we leave the configuration mode of the Fast Ethernet interface by typing `exit`. Similarly we enter the configuration mode of the serial interface by typing `interface Serial0/0` 2. We make this interface a PPP link by means of `encapsulation PPP` and assign a predefined IP address. Van Jacobson TCP header compression is enabled by `ip tcp header-compression`.

The definition of the internal address of the Cisco router is not as straightforward as on the IBM router, where you would simply use the `SET INTERNAL-ADDRESS` command. On Cisco routers the internal address of a router is derived as follows:

1. If there are any loopback interfaces, the router chooses the IP address of the loopback interface with the numerically highest IP address as the internal address.
2. If no loopback interfaces exist, the router chooses the numerically highest IP address of all configured interfaces as the internal address.

Thus the loopback interface definition in Figure 2 on page 17 serves to define the internal address of the router: 10.1.255.32/32. 3

The command `ip routing` is used to enable the router to act as an IP router 4. It is not necessary to use this command to enable IP routing because IP routing is enabled by default; in fact the only time it is likely to be needed is the converse `no ip routing` command which would be used to configure a bridging environment in which routing is disabled. We have used the `ip routing` command in our examples even though it is not strictly necessary. To enable RIP on the router, enter `router rip` to start the subconfiguration dialog of the RIP process 5. On Cisco routers you do not have the same fine-grained control over which types of routes are sent and received over a particular interface by the RIP process as you do on an IBM router. The Cisco router always receives net, subnet, and host routes and sends net, subnet, host, static, and default routes. The `network 10.0.0.0` command activates RIP on all interfaces of the router that belong to the 10.0.0.0 net.

The integration of the Cisco router in our test network depicted in Figure 1 on page 15 works without any problems, as you can see from the IP routing table on Chicago-Cisco2621 displayed in Figure 3 on page 19. All subnetworks from its own autonomous system can be seen (10.1.3X.0/24 and 10.1.255.3X/32) as well as the routes to the subnets in the other autonomous system that uses OSPF as routing protocol.

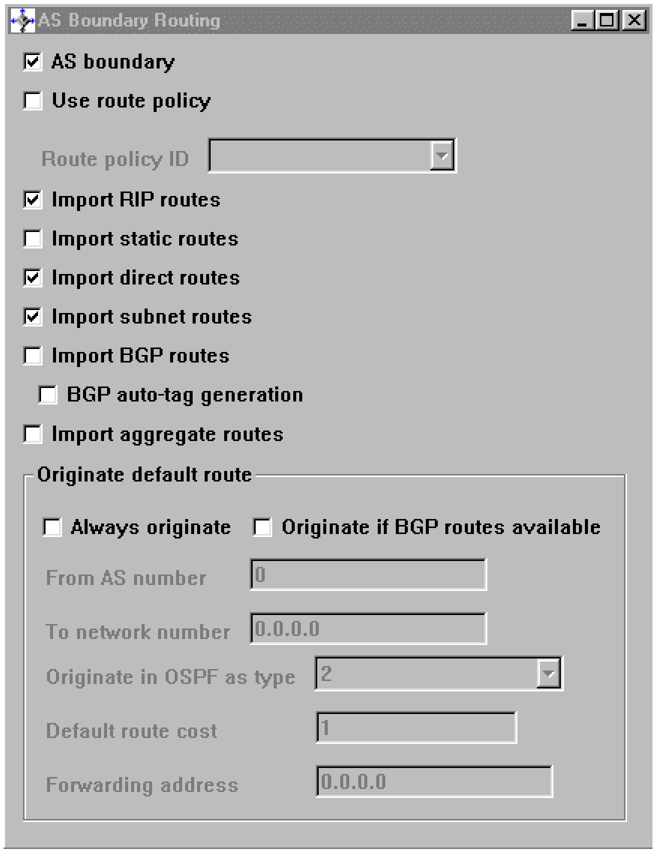
```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR, P - periodic downloaded static route
       T - traffic engineered route

Gateway of last resort is not set

        10.0.0.0/8 is variably subnetted, 28 subnets, 2 masks
R       10.1.4.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.5.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.6.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.1/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.2/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.3/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.20/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.11.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.13.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.14.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.2.2.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.30.0/24 [120/1] via 10.1.34.30, Serial0/0
R       10.1.31.0/24 [120/1] via 10.1.34.30, Serial0/0
R       10.1.32.0/24 [120/1] via 10.1.34.30, Serial0/0
R       10.1.33.0/24 [120/2] via 10.1.34.30, Serial0/0
C       10.1.34.0/24 is directly connected, Serial0/0
C       10.1.35.0/24 is directly connected, FastEthernet0/0
R       10.1.255.1/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.2/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.3/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.10/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.11/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.20/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.30/32 [120/1] via 10.1.34.30, Serial0/0
R       10.1.255.31/32 [120/2] via 10.1.34.30, Serial0/0
C       10.1.255.32/32 is directly connected, Loopback0
R       10.1.255.60/32 [120/2] via 10.1.34.30, Serial0/0
```

Figure 3. Routing table of Chicago-Cisco2621

The central Raleigh-IBM2216 router acts as an autonomous system border router that redistributes routes learned from OSPF into the RIP system and vice versa. This is done in the AS Boundary Routing panel of the configuration tool for MAS 3.4, which is shown in Figure 4 on page 20. Besides enabling AS Boundary Routing, we also have to import subnet routes into the OSPF area, since the subnets of the 10.0.0.0 network are spread over both autonomous systems. The import of direct routes is only necessary if the route to subnet 10.1.30.0/24 (the PPP link between Raleigh and Boston in Figure 1 on page 15) must be seen by other routers in the OSPF system, since from the point of view of Raleigh-IBM2216 this is not a RIP route but a direct route and would therefore not be redistributed without explicitly enabling it.



AS Boundary Routing

☑ AS boundary

☐ Use route policy

Route policy ID

☑ Import RIP routes

☐ Import static routes

☑ Import direct routes

☑ Import subnet routes

☐ Import BGP routes

☐ BGP auto-tag generation

☐ Import aggregate routes

Originate default route

☐ Always originate ☐ Originate 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

Forwarding address 0.0.0.0

Figure 4. Configuring AS boundary routing on Raleigh-IBM2216

2.3 Adding a Cisco router to an OSPF network

This section describes how to integrate Cisco routers in existing OSPF networks. For the WAN connectivity, we use a frame relay PVCs with – unless stated otherwise – the following parameters:

- *CIR* = 16x1024; a committed information rate of 16 kbps
- B_c = 16x1024; a committed burst size of 16 kb
- B_e = 4x1024; an excess burst size of 4 kb

As you can see from Figure 1 on page 15, the routers in OSPF area 0.0.0.1 are fully meshed. The OSPF network type of the 10.1.4.0/24 network is therefore non-broadcast multiple access (NBMA). Area 0.0.0.1 is also a stub area; that is, no external routes from other autonomous systems, in particular the RIPv2 system, are advertised in this area. Instead, a default route to the router that can reach the external system is advertised.

Each branch router in OSPF area 0.0.0.2 is connected to the central Raleigh-IBM2216 by a single PVC. From the point of view of OSPF, this network is treated as a point to multipoint (P2MP) network.

2.3.1 Adding a Cisco router to an NBMA network

To show how to add a Cisco router to an OSPF NBMA network consisting of a full mesh of FR PVCs, we replace the IBM router Sydney-IBM2210 by Sydney-Cisco2621. Figure 5 on page 22 shows the configuration dialog for Sydney-Cisco2621 that produces a configuration similar to that of Sydney-IBM2210.

```
Sydney-Cisco2621(config)#interface FastEthernet0/0
Sydney-Cisco2621(config-if)#speed auto
Sydney-Cisco2621(config-if)#ip address 10.1.5.10 255.255.255.0
Sydney-Cisco2621(config-if)#no shutdown
Sydney-Cisco2621(config-if)#exit
Sydney-Cisco2621(config-if)#!
Sydney-Cisco2621(config)#interface Serial0/0
Sydney-Cisco2621(config-if)#encapsulation frame-relay IETF     1
Sydney-Cisco2621(config-if)#frame-relay lmi-type ansi          2
Sydney-Cisco2621(config-if)#mtu 2048                           3
Sydney-Cisco2621(config-if)#ip address 10.1.4.10 255.255.255.0
Sydney-Cisco2621(config-if)#ip ospf network non-broadcast      4
Sydney-Cisco2621(config-if)#ip ospf hello-interval 10          5
Sydney-Cisco2621(config-if)#no shutdown
Sydney-Cisco2621(config-if)#exit
Sydney-Cisco2621(config-if)#!
Sydney-Cisco2621(config)#interface Loopback0
Sydney-Cisco2621(config-if)#ip address 10.1.255.10 255.255.255.255
Sydney-Cisco2621(config-if)#exit
Sydney-Cisco2621(config)#!
Sydney-Cisco2621(config)#ip routing
Sydney-Cisco2621(config)#!
Sydney-Cisco2621(config)#router ospf 1                        6
Sydney-Cisco2621(config-rou)#area 0.0.0.1 stub                7
Sydney-Cisco2621(config-rou)#network 10.1.4.0 0.0.0.255 area 0.0.0.1 8
Sydney-Cisco2621(config-rou)#network 10.1.5.0 0.0.0.255 area 0.0.0.1
Sydney-Cisco2621(config-rou)#network 10.1.255.10 0.0.0.0 area 0.0.0.1
Sydney-Cisco2621(config-rou)#exit
Sydney-Cisco2621(config)#end
Sydney-Cisco2621#
```

Figure 5. Configuring a Cisco router as part of an OSPF NBMA network

The configuration of the Fast Ethernet interface is as described in 2.2, “Adding a Cisco router to a RIP network” on page 16. The configuration of the serial interface that connects the router to the NBMA network is more complex than the PPP link we configured last time:

- The serial interface is configured to be a frame relay interface. Be aware that you have to specify the encapsulation technique explicitly to be RFC1490-compliant by adding the keyword `IETF` **1**. By default, the Cisco routers use a proprietary encapsulation technique that will not be recognized by IBM routers.
- Set the Local Management Interface (LMI) type to the one offered by the service provider **2**. Note that `ansi` is not the default.
- The default value of the maximum transmission unit (MTU) of serial interfaces is 1500 bytes on Cisco routers, which differs from 2048 bytes on

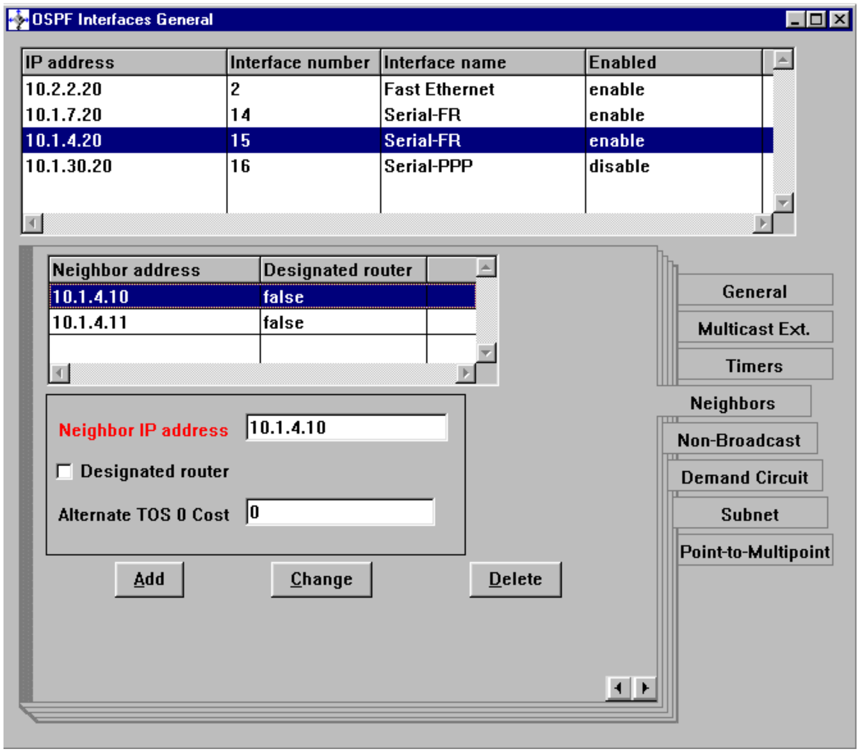
IBM routers. We adjust the MTU of the Cisco router to 2048 bytes; OSPF requires that the MTU sizes match3.

- After assigning an IP address to the serial interface, we tell the OSPF process to treat this interface as a non-broadcast multiple access network 4.
- The default value on both Cisco and IBM routers for the hello interval is 10 seconds. The hello interval is included in the hello packets OSPF routers exchange on a common subnet and when received tested for equality with the receiving router's hello interval. If it does not match, the routers cannot become neighbors. Note that RFC 2328[3] seems to advise that the hello interval be 30 seconds for serial interfaces; both IBM and Cisco default to 10 seconds for all interfaces but it will certainly do no harm to specify it explicitly. Likewise, the dead router interval of all network interfaces running OSPF on a subnet must match; the dead router intervals default on both Cisco and IBM routers to four times the configured hello interval and therefore match by default if the hello intervals already match. 5
- The OSPF routing process is started by the command `router ospf 1`. The number serves to distinguish different OSPF routing processes on a Cisco router and has no relevance outside the router[4]. 6
- The `area stub` command tells the OSPF process to treat OSPF area 0.0.0.1 as a stub area 7.
- Next, the interfaces are assigned to OSPF areas by means of the `network area` command 8. The Cisco router tests each interface whether it is in the range specified by each `network` *`<network address> <inverse mask>`* `area` *`<area number>`* command. The second number in the command has to be interpreted as an inverse mask to be applied to the first number. All interfaces that have an IP address in the range of addresses described by the pair of network address and inverse network mask are assigned to the given OSPF area.

In a router on at least one side of each PVC in an NBMA network, the neighbor routers have to be defined. To simplify the configuration of the branch routers we defined the neighbors on the correspondent frame relay interface of the central IBM2216, as depicted in Figure 6 on page 24. The 2216 itself also has to be defined as a designated router for this OSPF area in this case (since neither of the other routers are so defined).

[3] RFC2328: OSPF Version 2; the values given for the hello interval in the RFC are little more than a suggestion.

[4] Note that in an autonomous system using the (E)IGRP, *<number>* in the `router (e)igrp` *`<number>`* command denotes the autonomous system number and must be identical throughout the autonomous system.



IP address	Interface number	Interface name	Enabled
10.2.2.20	2	Fast Ethernet	enable
10.1.7.20	14	Serial-FR	enable
10.1.4.20	15	Serial-FR	enable
10.1.30.20	16	Serial-PPP	disable

Neighbor address	Designated router
10.1.4.10	false
10.1.4.11	false

Neighbor IP address 10.1.4.10

□ Designated router

Alternate TOS 0 Cost 0

Figure 6. Neighbor definition for an NBMA network on Raleigh-IBM2216

The routing table of Sydney-Cisco2621 is shown in Figure 7. As we expected, external routes to the autonomous system running RIPv2 are not advertised in the stub area 0.0.0.1. Instead, the router learned a default route (see the entry for `Gateway of last resort`) pointing to the autonomous system border router.

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 10.1.4.20 to network 0.0.0.0

        10.0.0.0/8 is variably subnetted, 18 subnets, 2 masks
O IA    10.1.11.0/24 [110/3134] via 10.1.4.20, 00:04:03, Serial0/0
O IA    10.1.13.0/24 [110/3130] via 10.1.4.20, 00:04:03, Serial0/0
O IA    10.1.14.0/24 [110/3134] via 10.1.4.20, 00:04:03, Serial0/0
O IA    10.2.2.0/24 [110/1572] via 10.1.4.20, 00:04:03, Serial0/0
O IA    10.1.7.1/32 [110/3124] via 10.1.4.20, 00:04:03, Serial0/0
O       10.1.6.0/24 [110/1572] via 10.1.4.11, 00:04:03, Serial0/0
O IA    10.1.7.2/32 [110/3124] via 10.1.4.20, 00:04:03, Serial0/0
C       10.1.5.0/24 is directly connected, FastEthernet0/0
O IA    10.1.7.3/32 [110/3124] via 10.1.4.20, 00:04:03, Serial0/0
C       10.1.4.0/24 is directly connected, Serial0/0
O IA    10.1.7.20/32 [110/1562] via 10.1.4.20, 00:04:03, Serial0/0
O IA    10.1.255.60/32 [110/1572] via 10.1.4.20, 00:04:06, Serial0/0
O       10.1.255.20/32 [110/1562] via 10.1.4.20, 00:04:06, Serial0/0
O IA    10.1.255.1/32 [110/3124] via 10.1.4.20, 00:04:06, Serial0/0
O IA    10.1.255.2/32 [110/3124] via 10.1.4.20, 00:04:06, Serial0/0
O IA    10.1.255.3/32 [110/3125] via 10.1.4.20, 00:04:06, Serial0/0
C       10.1.255.10/32 is directly connected, Loopback0
O       10.1.255.11/32 [110/1562] via 10.1.4.11, 00:04:06, Serial0/0
O*IA    0.0.0.0/0 [110/1563] via 10.1.4.20, 00:04:06, Serial0/0
```

Figure 7. Routing table of Sydney-Cisco2621

2.3.2 Adding a Cisco router to a P2MP network

The insertion of a Cisco router into a OSPF P2MP network resembles the integration into a NBMA network, as can be seen from Figure 8 on page 26.

```
Bonn-Cisco2621(config)#interface FastEthernet0/0
Bonn-Cisco2621(config-if)#speed auto
Bonn-Cisco2621(config-if)#ip address 10.1.11.3 255.255.255.0
Bonn-Cisco2621(config-if)#no shutdown
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config-if)#!
Bonn-Cisco2621(config)#interface Serial0/0
Bonn-Cisco2621(config-if)#encapsulation frame-relay IETF
Bonn-Cisco2621(config-if)#frame-relay lmi-type ansi
Bonn-Cisco2621(config-if)#mtu 2048
Bonn-Cisco2621(config-if)#ip address 10.1.7.3 255.255.255.0
Bonn-Cisco2621(config-if)#ip ospf network point-to-multipoint 1
Bonn-Cisco2621(config-if)#ip ospf hello-interval 10
Bonn-Cisco2621(config-if)#no shutdown
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config-if)#!
Bonn-Cisco2621(config)#interface Loopback0
Bonn-Cisco2621(config-if)#ip address 10.1.255.3 255.255.255.255
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#ip routing
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#router ospf 2                          2
Bonn-Cisco2621(config-rou)#network 10.1.7.0 0.0.0.255 area 0.0.0.2
Bonn-Cisco2621(config-rou)#network 10.1.11.0 0.0.0.255 area 0.0.0.2
Bonn-Cisco2621(config-rou)#network 10.1.255.3 0.0.0.0 area 0.0.0.2
Bonn-Cisco2621(config-rou)#exit
Bonn-Cisco2621(config)#end
Bonn-Cisco2621#
```

Figure 8. Configuring a Cisco router as part of an OSPF P2MP network

In fact, the only difference is the declaration that the frame relay cloud should be viewed as a point-to-multipoint network 1. We choose “2” instead of “1” 2 as the process ID of the `router ospf` command simply to demonstrate that this value is local to the router. As in 2.3.1, “Adding a Cisco router to an NBMA network” on page 21, neighbors only have to be defined on one end of the PVC, which we have done on the central IBM2210. The fact that this is a simpler configuration task than for an NMBA network is not seen here.

Unlike area 0.0.0.1, area 0.0.0.2 is not a stub area, thus external routes from the RIPv2 autonomous system should be seen in the routing table of Bonn-Cisco2621 – and indeed, in Figure 9 on page 27 you can see (beginning at 4) the six routes to the subnets in the RIPv2 system, and (beginning at 5) the three host routes.

The routes at 1 and 2 also serve to show the difference between routing within an NBMA cloud and within a P2MP cloud. In a P2MP network, each router must have an explicit host route to each other router on the P2MP subnet; that is, 10.1.7.1 and 10.1.7.2 (Munich and Berlin) must show up in Bonn’s routing table as routed via the center (Raleigh, 10.1.7.20) of the P2MP network. Compare this with the routing table for the Sydney 2621 (Figure 7 on page 25) in which a single network route (10.1.4.0/24) suffices and, specifically, there is no comparable host route for Melbourne (which would be 10.1.4.11/32).

```
iCodes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR, P - periodic downloaded static route
       T - traffic engineered route

Gateway of last resort is not set

        10.0.0.0/8 is variably subnetted, 25 subnets, 2 masks
C       10.1.11.0/24 is directly connected, FastEthernet0/0
O       10.1.14.0/24 [110/3134] via 10.1.7.20, Serial0/0
O       10.1.13.0/24 [110/3130] via 10.1.7.20, Serial0/0
C       10.1.7.0/24 is directly connected, Serial0/0
O       10.1.7.1/32 [110/3124] via 10.1.7.20, Serial0/0        1
O       10.1.7.2/32 [110/3124] via 10.1.7.20, Serial0/0        2
O       10.1.7.20/32 [110/1562] via 10.1.7.20, Serial0/0
O IA    10.1.4.0/24 [110/3124] via 10.1.7.20, Serial0/0        3
O IA    10.1.5.0/24 [110/3134] via 10.1.7.20, Serial0/0
O IA    10.1.6.0/24 [110/3134] via 10.1.7.20, Serial0/0
O E2    10.1.30.0/24 [110/1] via 10.1.7.20, Serial0/0          4
O E2    10.1.31.0/24 [110/2] via 10.1.7.20, Serial0/0
O E2    10.1.32.0/24 [110/2] via 10.1.7.20, Serial0/0
O E2    10.1.34.0/24 [110/2] via 10.1.7.20, Serial0/0
O E2    10.1.35.0/24 [110/3] via 10.1.7.20, Serial0/0
O E2    10.1.33.0/24 [110/3] via 10.1.7.20, Serial0/0
O IA    10.2.2.0/24 [110/1572] via 10.1.7.20, Serial0/0
O IA    10.1.255.60/32 [110/1572] via 10.1.7.20, Serial0/0
O       10.1.255.1/32 [110/3124] via 10.1.7.20, Serial0/0
O       10.1.255.2/32 [110/3124] via 10.1.7.20, Serial0/0
C       10.1.255.3/32 is directly connected, Loopback0
O IA    10.1.255.10/32 [110/3124] via 10.1.7.20, Serial0/0
O IA    10.1.255.11/32 [110/3124] via 10.1.7.20, Serial0/0
O       10.1.255.20/32 [110/1562] via 10.1.7.20, Serial0/0
O E2    10.1.255.30/32 [110/2] via 10.1.7.20, Serial0/0        5
O E2    10.1.255.31/32 [110/3] via 10.1.7.20, Serial0/0
O E2    10.1.255.32/32 [110/3] via 10.1.7.20, Serial0/0
```

Figure 9. Routing table of Bonn-Cisco2621

2.4 Adding an EIGRP system to an IBM OSPF system

In this section we study the case where there is already a network of Cisco routers running the Enhanced Interior Gateway Routing Protocol (EIGRP) that has to integrated into an existing IBM OSFP/RIP network. In 2.4.1, “Configuring EIGRP” on page 28, we show how EIGRP is configured in the network that has to be integrated, in 2.4.2, “Redistributing EIGRP and OSPF” on page 32, the details of importing/exporting the routes from/to the existing network are described.

This case study obviously differs from one in which a single Cisco router is added to an existing IBM router network, because in such an environment either OSPF or RIP would be used on the Cisco router. The sort of network we are thinking about here is one that might result from a corporate merger in which two totally different networks - one IBM and one Cisco - are to be combined.

2.4.1 Configuring EIGRP

In Figure 10 on page 29 you can see a small Cisco EIGRP network with five subnets of 10.0.0.0 that we are going to attach to our existing base IBM RIP/OSPF network. Note that – unlike with OSPF – EIGRP cannot treat the frame relay cloud as a single IP network, neither as NBMA nor P2MP. Each PVC is treated as a single subnet. To preserve IP addresses you should use smaller subnets on the link or use an unnumbered link. We used the subnets 10.1.40.0/24 and 10.1.41.0/24, since our IP address scheme is for test purposes only and optimized for convenience and clarity rather than for efficiency of IP address utilization.

EIGRP is aware of the autonomous system number of the system it runs in. We choose 20 as the number of the small autonomous system shown in Figure 10 on page 29. Within an autonomous system, EIGRP has a flat topology, in contrast to OSPF with its hierarchical areas. EIGRP is a distance/vector protocol, like RIP and unlike OSPF, but one that has been optimized to give many of the efficiencies of OSPF.

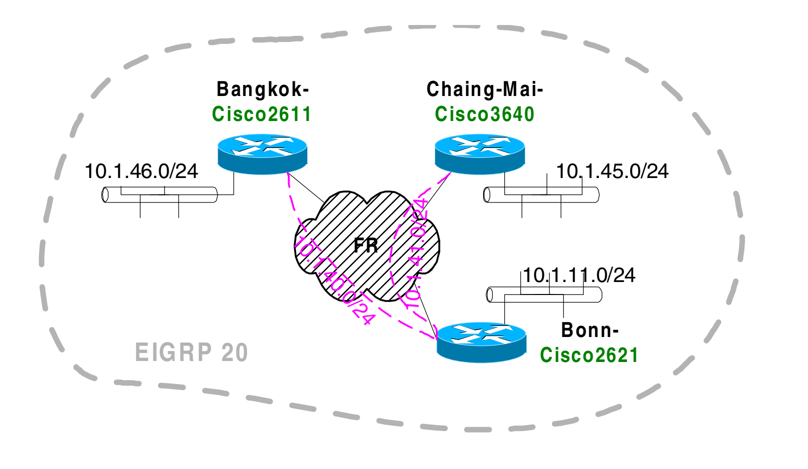


Figure 10. Small Cisco network using EIGRP

Figure 11 on page 30 shows the configuration of Bangkok-Cisco2611 as member of the EIGRP system. The configuration of the interfaces is straightforward; to enable EIGRP, you enter the `router eigrp` *`<autonomous system number>`* **1** command. The autonomous system number has to be identical on all routers in the same EIGRP system. You then specify the interfaces that should be part of the EIGRP system by means of the `network` command **2**. The technique is the same as for the `network area` command for the OSPF configuration. You specify a network number and an *inverse* mask. The interfaces whose addresses are in the range specified by one of the `network` statements participate in the EIGRP process.

```
Bangkok-Cisco2611(config)#interface Ethernet0/0
Bangkok-Cisco2611(config-if)#ip address 10.1.46.42 255.255.255.0
Bangkok-Cisco2611(config-if)#no shutdown
Bangkok-Cisco2611(config-if)#exit
Bangkok-Cisco2611(config-if)#!
Bangkok-Cisco2611(config)#interface Serial0/0
Bangkok-Cisco2611(config-if)#encapsulation frame-relay IETF
Bangkok-Cisco2611(config-if)#frame-relay lmi-type ansi
Bangkok-Cisco2611(config-if)#mtu 2048
Bangkok-Cisco2611(config-if)#ip address 10.1.40.42 255.255.255.0
Bangkok-Cisco2611(config-if)#no shutdown
Bangkok-Cisco2611(config-if)#exit
Bangkok-Cisco2611(config-if)#!
Bangkok-Cisco2611(config)#interface Loopback0
Bangkok-Cisco2611(config-if)#ip address 10.1.255.42 255.255.255.255
Bangkok-Cisco2611(config-if)#exit
Bangkok-Cisco2611(config)#!
Bangkok-Cisco2611(config)#ip routing
Bangkok-Cisco2611(config)#!
Bangkok-Cisco2611(config)#router eigrp 20                   1
Bangkok-Cisco2611(config-rou)#network 10.1.40.0 0.0.0.255    2
Bangkok-Cisco2611(config-rou)#network 10.1.46.0 0.0.0.255
Bangkok-Cisco2611(config-rou)#network 10.1.255.42 0.0.0.0
Bangkok-Cisco2611(config-rou)#exit
Bangkok-Cisco2611(config)#end
Bangkok-Cisco2611#
```

Figure 11. Configuring EIGRP on Bangkok-Cisco2611

The configuration of Chaing-Mai-Cisco3640 is analogous to Bangkok-Cisco2611.

We show the configuration of Bonn-Cisco2621 in more detail in Figure 12 on page 31, since it uses *subinterfaces* on its frame relay interface:

```
Bonn-Cisco2621(config)#interface FastEthernet0/0
Bonn-Cisco2621(config-if)#speed 10
Bonn-Cisco2621(config-if)#ip address 10.1.11.3 255.255.255.0
Bonn-Cisco2621(config-if)#no shutdown
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config-if)#!
Bonn-Cisco2621(config)#interface Serial0/0
Bonn-Cisco2621(config-if)#encapsulation frame-relay IETF        1
Bonn-Cisco2621(config-if)#frame-relay lmi-type ansi
Bonn-Cisco2621(config-if)#mtu 2048
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config-if)#!
Bonn-Cisco2621(config)#interface Serial0/0.1 point-to-point     2
Bonn-Cisco2621(config-subif)#ip address 10.1.40.3 255.255.255.0
Bonn-Cisco2621(config-subif)#frame-relay interface-dlci 167      3
Bonn-Cisco2621(config-fr-dlci)#exit
Bonn-Cisco2621(config-subif)#exit
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#interface Serial0/0.2 point-to-point
Bonn-Cisco2621(config-subif)#ip address 10.1.41.3 255.255.255.0
Bonn-Cisco2621(config-subif)#frame-relay interface-dlci 168
Bonn-Cisco2621(config-fr-dlci)#exit
Bonn-Cisco2621(config-subif)#exit
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#interface Serial0/0
Bonn-Cisco2621(config-if)#no shutdown
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#interface Loopback0
Bonn-Cisco2621(config-if)#ip address 10.1.255.3 255.255.255.255
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#ip routing
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#router eigrp 20                        4
Bonn-Cisco2621(config-rou)#network 10.1.40.0 0.0.0.255
Bonn-Cisco2621(config-rou)#network 10.1.41.0 0.0.0.255
Bonn-Cisco2621(config-rou)#network 10.1.11.0 0.0.0.255
Bonn-Cisco2621(config-rou)#network 10.1.255.3 0.0.0.0
Bonn-Cisco2621(config-rou)#exit
Bonn-Cisco2621(config)#end
Bonn-Cisco2621#
```

Figure 12. Configuring EIGRP on Bonn-Cisco2621

- The encapsulation technique, the LMI type and the MTU apply to the interface as a whole 1.

- Then you define a subinterface and associate the corresponding FR PVC with it 2, 3. The `point-to-point` command shows that – if we were using OSPF – you could treat the link as a point-to-multipoint link. Of course, you would have to add more `frame-relay interface-dlci` statements in this case.
- The `router eigrp` command starts the EIGRP process on the router. Note that the identical autonomous system is 20, that is, the same as on Chaing-Mai-Cisco3640 and Bangkok-Cisco2611.

Figure 13 gives you an idea how the routing table looks on Chaing-Mai-Cisco3640 in this small EIGRP network, at this point with no additional attachment to our base network.

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
       U - per-user static route, o - ODR

Gateway of last resort is not set

        10.0.0.0/8 is variably subnetted, 8 subnets, 2 masks
D       10.1.11.0/24 [90/1787392] via 10.1.41.3, 00:02:06, Serial0/0
C       10.1.41.0/24 is directly connected, Serial0/0
C       10.1.45.0/24 is directly connected, Ethernet0/0
D       10.1.40.0/24 [90/2273792] via 10.1.41.3, 00:02:06, Serial0/0
D       10.1.46.0/24 [90/2299392] via 10.1.41.3, 00:02:06, Serial0/0
C       10.1.255.41/32 is directly connected, Loopback0
D       10.1.255.42/32 [90/2401792] via 10.1.41.3, 00:02:06, Serial0/0
D       10.1.255.3/32 [90/1889792] via 10.1.41.3, 00:02:06, Serial0/0
```

Figure 13. Routing table of Chaing-Mai-Cisco2621 in the EIGRP system

2.4.2 Redistributing EIGRP and OSPF

Now we attach the EIGRP network of Figure 10 on page 29 to our basic network by replacing Bonn-IBM2210 with Bonn-Cisco2621 and its EIGRP "appendix". The resulting network is depicted in Figure 14 on page 33. Moreover, we added a data center router (Raleigh-Cisco7505) to the Ethernet LAN on the Raleigh campus just to prove that the OSPF implementations of IBM and Cisco interoperate without any particular configuration settings on a LAN.

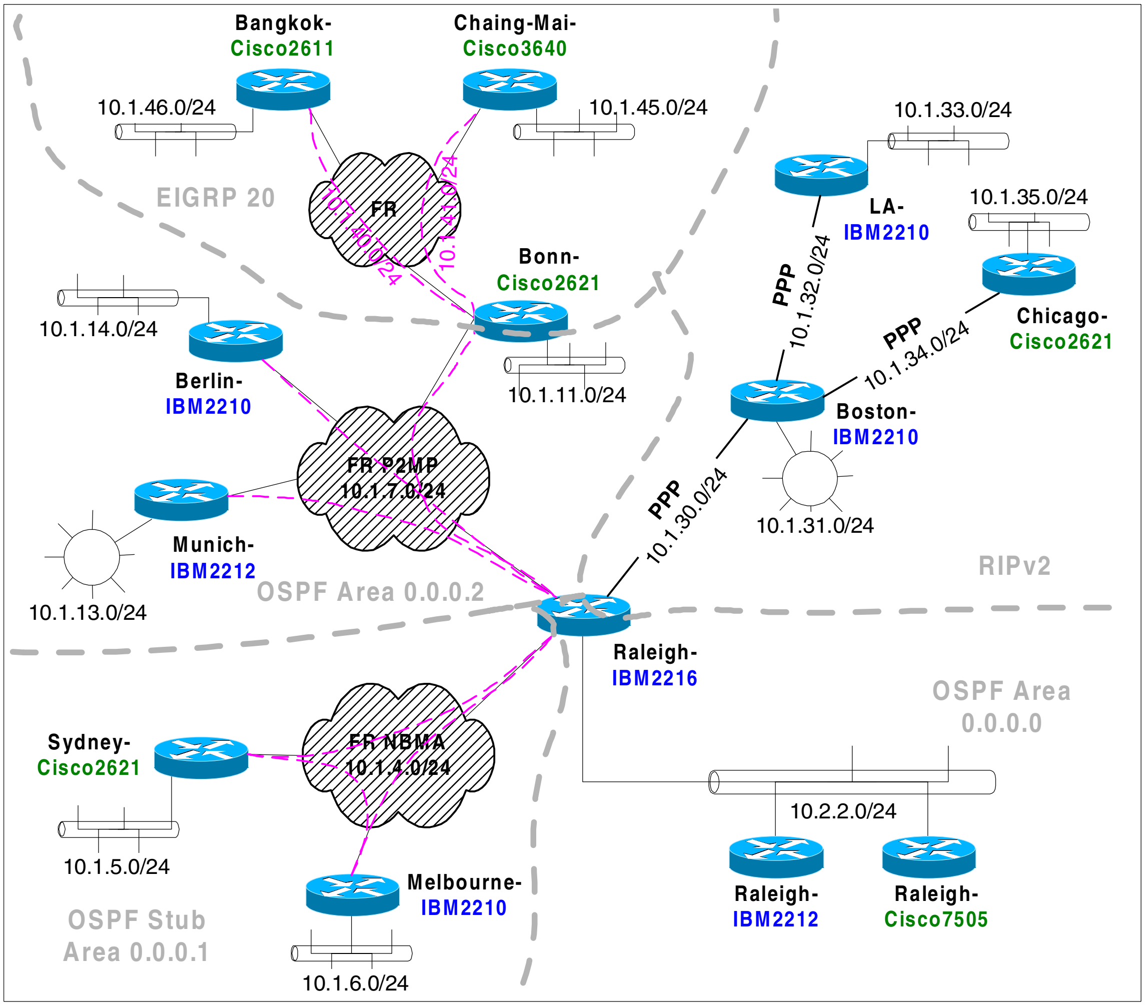


Figure 14. Overview of the combined EIGRP/RIP/OSPF network

In the following we show how the Bonn-Cisco2621 must be configured in order to act as an autonomous system border router between the EIGRP and the OSPF systems. The configuration dialog in Figure 15 on page 34 skips the interface configurations that have already been shown in Figure 12 on page 31. We now focus on the definition of an additional sub-interface to connect Bonn-Cisco2621 to the P2MP FR cloud and the route redistribution commands:

```
Bonn-Cisco2621(config)#interface Serial0/0.3 multipoint        1
Bonn-Cisco2621(config-if)#ip address 10.1.7.3 255.255.255.0
Bonn-Cisco2621(config-if)#ip ospf network point-to-multipoint
Bonn-Cisco2621(config-if)#ip ospf hello-interval 10
Bonn-Cisco2621(config-if)#frame-relay interface-dlci 172
Bonn-Cisco2621(config-fr-dlci)#exit
Bonn-Cisco2621(config-if)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#router eigrp 20
Bonn-Cisco2621(config-rou)#no network 10.1.11.0 0.0.0.255     2
Bonn-Cisco2621(config-rou)#no network 10.1.255.3 0.0.0.0      3
Bonn-Cisco2621(config-rou)#redistribute ospf 1 metric 64 100 255
                        40 2048                               4
Bonn-Cisco2621(config-rou)#exit
Bonn-Cisco2621(config)#!
Bonn-Cisco2621(config)#router ospf 1
Bonn-Cisco2621(config-rou)#redistribute eigrp 20 metric 2000
                        metric-type 1 subnets                 5
Bonn-Cisco2621(config-rou)#network 10.1.7.0 0.0.0.255 area 0.0.0.2
Bonn-Cisco2621(config-rou)#network 10.1.11.0 0.0.0.255 area 0.0.0.2
Bonn-Cisco2621(config-rou)#network 10.1.255.3 0.0.0.0 area 0.0.0.2
Bonn-Cisco2621(config-rou)#exit
Bonn-Cisco2621(config)#end
Bonn-Cisco2621#
```

Figure 15. Configuring redistribution on Bonn-Cisco2621

- First we define an additional subinterface that attaches Bonn-Cisco2621 to the P2MP network 1.
- Then we take the interfaces 10.1.10.3/24 and 10.1.255.3/32 out of the EIGRP system, subsequently to add them to the OSPF area 0.0.0.2 2,3
- We tell the EIGRP 20 process to redistribute routes from the OSPF 1 process. These learned routes must be fit with a metric that is understood in the EIGRP system. EIGRP metrics are calculated from a 5-tuple that consists of {bandwidth [kbps], delay [multiple of 10µs], reliability [0..255], load [0..255], path MTU [bytes]}. After the keyword `metric` in the `redistribute` command we define the 5-tuple {64,100,255,40,2048} for the EIGRP metric for all routes that are imported from OSPF. 4
- Similarly, the OSPF process redistributes routes of the EIGRP process 5. The OSPF metric associated with all learned EIGRP routes is 2000. Furthermore, `metric-type 1` tells the OSPF to advertise the EIGRP routes as OSPF external routes of type 1. This is important, since we noticed the Raleigh-IBM2216 router redistributes only OSPF external routes of type 1 into other autonomous systems, in our case the RIPv2 system. Finally, `subnets` tells the OSPF process to import subnet routes, not only genuine

network routes. This is important, since otherwise only a route to the network 10.0.0.0/8 would be imported from the EIGRP process.

Comparing the redistribution process on IBM and Cisco routers, we found that on the Cisco routers the redistribution can be adjusted better. On the IBM routers it is not possible to control the metrics of imported/exported routes, which can be a problem when metrics are not comparable, as is the case for the hop count metric of the RIPv2 system and the metric of the OSPF system. If routes are not comparable, they must be imported into the OSPF system as OSPF external routes type 2. The metrics of these routes remain unchanged while being distributed in the OSPF system, whereas external routes of type 1 are dealt with as “normal” routes, that is their metric is increased along their path. This can be observed in the routing table of Berlin-IBM2210 displayed in Figure 16 on page 36.

```
Type    Dest net          Mask      Cost     Age       Next hop(s)

SPE1    10.0.0.0          FF000000  5124     50        10.1.7.20
SPIA    10.1.4.0          FFFFFF00  3124     12632     10.1.7.20
SPIA    10.1.5.0          FFFFFF00  3125     477       10.1.7.20
SPIA    10.1.6.0          FFFFFF00  3134     12592     10.1.7.20
 Dir*   10.1.7.0          FFFFFF00  1        783874    FR/0
 SPF    10.1.7.1          FFFFFFFF  0        783874    FR/0
 SPF    10.1.7.2          FFFFFFFF  3124     12617     10.1.7.20
 SPF    10.1.7.3          FFFFFFFF  3124     5750      10.1.7.20
 SPF    10.1.7.20         FFFFFFFF  1562     12632     10.1.7.20
 SPF    10.1.11.0         FFFFFF00  3134     5744      10.1.7.20
 SPF    10.1.13.0         FFFFFF00  3130     12617     10.1.7.20
SPE2    10.1.30.0         FFFFFF00  1        12632     10.1.7.20
SPE2    10.1.31.0         FFFFFF00  2        12632     10.1.7.20
SPE2    10.1.32.0         FFFFFF00  2        12632     10.1.7.20
SPE2    10.1.33.0         FFFFFF00  3        12632     10.1.7.20 1
SPE2    10.1.34.0         FFFFFF00  2        12632     10.1.7.20
SPE2    10.1.35.0         FFFFFF00  3        12632     10.1.7.20
SPE1    10.1.40.0         FFFFFF00  5124     1222      10.1.7.20
SPE1    10.1.41.0         FFFFFF00  5124     1222      10.1.7.20
SPE1    10.1.46.0         FFFFFF00  5124     157       10.1.7.20 2
 SPF*   10.1.255.1        FFFFFFFF  0        784174    SINK/0
 SPF    10.1.255.2        FFFFFFFF  3124     12620     10.1.7.20
 SPF    10.1.255.3        FFFFFFFF  3125     3946      10.1.7.20
SPIA    10.1.255.10       FFFFFFFF  3125     480       10.1.7.20
SPIA    10.1.255.11       FFFFFFFF  3124     12594     10.1.7.20
 SPF    10.1.255.20       FFFFFFFF  1562     12635     10.1.7.20
SPE2    10.1.255.30       FFFFFFFF  2        12635     10.1.7.20
SPE2    10.1.255.31       FFFFFFFF  3        12635     10.1.7.20
SPE2    10.1.255.32       FFFFFFFF  3        12635     10.1.7.20
SPE1    10.1.255.41       FFFFFFFF  5124     79        10.1.7.20
SPE1    10.1.255.42       FFFFFFFF  5124     159       10.1.7.20
SPIA    10.1.255.50       FFFFFFFF  1573     12635     10.1.7.20 3
SPIA    10.1.255.60       FFFFFFFF  1572     12635     10.1.7.20
SPIA    10.2.2.0          FFFFFF00  1572     12635     10.1.7.20
```

Figure 16. Routing table of Berlin-IBM2210 in the combined network

As an example for an external route of type 2 take the network route to Ethernet 10.1.33.0/24 (behind LA-IBM2210) 1. Its metric is (hop count) 3, which is the distance from the autonomous system border router to the network in the RIPv2 system. Throughout the OSPF system (apart from stub area 0.0.0.1 of course) this route will be advertised with metric 3. Compare this with the network route to Ethernet 10.1.46.0/24 (behind Bangkok-Cisco2611) 2. Its metric is 5124 = 2000 + 1562 + 1562. This is due to the fact that external routes of type 1 accumulate the metrics along their path: the sum of 2000 (added to all redistributed EIGRP routes) and 2x1562 for the two hops over 64 kbps frame relay interfaces.

To see how the redistributed routes look in the RIPv2 system, see Figure 18 on page 38, which shows the routing table of Chicago-Cisco2621. A routing table from a router in the EIGRP system is shown in Figure 17 on page 37.

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - default
       U - per-user static route, o - ODR

Gateway of last resort is not set

        10.0.0.0/8 is variably subnetted, 34 subnets, 2 masks
D EX    10.1.11.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.13.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.2.2.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.7.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.7.1/32 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.6.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.5.0/24 [170/40537600] via 10.1.41.3, 00:03:06, Serial0/0
D EX    10.1.7.2/32 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.4.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.31.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.30.0/24 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.7.20/32 [170/40537600] via 10.1.41.3, 00:03:08, Serial0/0
C       10.1.41.0/24 is directly connected, Serial0/0
D       10.1.40.0/24 [90/41024000] via 10.1.41.3, 00:03:11, Serial0/0
D       10.1.46.0/24 [90/41049600] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.35.0/24 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.34.0/24 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.33.0/24 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.32.0/24 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.50/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.60/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.32/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
C       10.1.255.41/32 is directly connected, Loopback0
D       10.1.255.42/32 [90/41152000] via 10.1.41.3, 00:03:08, Serial0/0
D EX    10.1.255.20/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.30/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.31/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.1/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.2/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D       10.1.255.3/32 [90/40640000] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.10/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
D EX    10.1.255.11/32 [170/40537600] via 10.1.41.3, 00:03:11, Serial0/0
```

Figure 17. Routing table of Chaing-Mai-Cisco3640 in the combined network

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - default
       U - per-user static route, o - ODR,
               P - periodic downloaded static route
       T - traffic engineered route

Gateway of last resort is not set

        10.0.0.0/8 is variably subnetted, 35 subnets, 2 masks
R       10.1.11.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.13.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.2.2.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.6.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.1/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.5.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.2/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.7.3/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.4.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.31.0/24 [120/1] via 10.1.34.30, Serial0/0
R       10.1.30.0/24 [120/1] via 10.1.34.30, Serial0/0
R       10.1.7.20/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.41.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.40.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.46.0/24 [120/2] via 10.1.34.30, Serial0/0
C       10.1.35.0/24 is directly connected, FastEthernet0/0
C       10.1.34.0/24 is directly connected, Serial0/0
R       10.1.33.0/24 [120/2] via 10.1.34.30, Serial0/0
R       10.1.32.0/24 [120/1] via 10.1.34.30, Serial0/0
R       10.1.255.50/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.60/32 [120/2] via 10.1.34.30, Serial0/0
C       10.1.255.32/32 is directly connected, Loopback0
R       10.1.255.41/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.42/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.20/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.30/32 [120/1] via 10.1.34.30, Serial0/0
R       10.1.255.31/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.1/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.2/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.3/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.10/32 [120/2] via 10.1.34.30, Serial0/0
R       10.1.255.11/32 [120/2] via 10.1.34.30, Serial0/0
```

Figure 18. Routing table of Chicago-Cisco2621 in the combined network

For the sake of completeness we add a table showing the type and software releases of all Cisco routers used in the combined network:

Table 8. Employed Cisco routers: model type, internal address, and software release

Router name	Internal address	Model	Software release
Bangkok-Cisco2611	10.1.255.42	2611	(C2600-JS-M), Version 12.0(4)T
Bonn-Cisco2621	10.1.255.3	2621	(C2600-JS-M), Version 12.0(7)T
Chaing-Mai-Cisco3640	10.1.255.41	3640	(C3640-JS-M), Version 12.0(6)
Chicago-Cisco2621	10.1.255.32	2621	(C2600-JS-M), Version 12.0(4)T
Raleigh-Cisco7505	10.1.255.50	7505	(RSP-JK2SV-M), Version 12.0(5)T1
Sydney-Cisco2621	10.1.255.10	4700	(C4500-JS40-M), Version 12.0(5)

Chapter 3. Voice over IP and voice over frame relay

The majority of existing IBM 2210/2212/2216 router networks have been designed to transport data. With increased emphasis on unified networks and the cost savings that can be achieved by carrying voice traffic via the data network, many organizations wish to add voice transport functions to existing IBM router networks.

Recent enhancements to the IBM router common code, specifically in Version 3.4, have been designed to support the transport of voice traffic across IBM router backbones. OEM routers and FRADs may be used to provide the direct attachment to the telephony devices and to digitize the voice signal for transport through the router network. A wide range of OEM router and FRAD equipment may be installed to provide direct interfaces to voice devices such as telephones, PABXs (private automatic branch telephone exchanges), and fax machines.

This chapter reviews the interoperability and migration issues that affect these organizations by adding voice-capable equipment to IBM router networks. Specific configurations supporting the addition of Cisco voice-capable routers to an IBM router network are reviewed.

Analog voice adapters are available for the IBM 2212 to support voice over frame relay (VoFR) implementations only. Cisco routers may be added to the network to provide a wide range of voice connectivity options supporting both Voice over IP (VoIP) and voice over frame relay.

The voice interface options available on Cisco routers are summarized in Table 9.

Table 9. Cisco router voice interface options

Voice interface	Cisco 1750	Cisco 2600	Cisco 3600	Cisco 7200
FXS	2 -4	2 - 4	2 - 16	N/A
FXO	2 - 4	2 - 4	2 - 16	N/A
E&M	2 - 4	2 - 4	2 - 16	N/A
ISDN BRI	N/A	2 - 4	2 - 16	N/A
E1/T1 Digital	N/A	2	12	2 - 12

Two primary options are available to allow the transport of digitally encoded voice traffic through a router wide area network: voice over frame relay (VoFR) and voice over IP (VoIP). VoFR encapsulates voice packets directly in frame relay frames with no layer 3 header, whereas VoIP encapsulates voice packets in IP packets using Real Time Protocol (RTP).

For more information on RTP and on how the most recent release of IBM router code can handle VoIP traffic see *Application-Driven Networking: Class of Service in IP, Ethernet and ATM Networks*, SG24-5384.

It is possible to add OEM routers to provide analog or digital telephony interfaces and to transport the voice traffic via a backbone IBM router network using either VoFR or VoIP. A number of techniques are employed to achieve efficiencies in voice transport over low-bandwidth WAN links. Some are proprietary; however, a number of standards are now in place that allow some degree of interoperability between routers in this area. These interoperability issues are addressed in the following sections.

3.1 Voice over IP

The transport of voice traffic via IP packets is a more recent development than VoFR. It is inherently more flexible as it is a layer 3 transport protocol and is therefore not tied to a single link protocol such as frame relay.

Voice over IP network links could include:

- PPP links using ISDN or dedicated digital services
- frame relay (in which IP traffic is configured to use frame relay links)
- LAN segments

Voice over IP devices include IP telephones that digitize voice and transfer it directly to an Ethernet LAN segment, but our network used more traditional telephones connected via voice adapter cards on Cisco routers.

VoIP is inherently less efficient/more wasteful of bandwidth than VoFR due to the additional frame headers required in each voice packet: first to encapsulate the voice in IP/UDP/RTP and then to encapsulate the IP packet in PPP, frame relay or as a LAN frame. However, a great deal of development has occurred to provide standards-based mechanisms to reduce this framing overhead and allow VoIP systems to provide similar efficiency to VoFR implementations. Many vendors, including IBM and Cisco, have also implemented extensive QoS and link efficiency features in their router platforms to support quality voice connections over IP routed networks.

3.1.1 VoIP network components

This section reviews the connectivity options and components required to allow voice traffic to be transported using VoIP in a wide area network environment. The configuration is shown in Figure 19 on page 43.

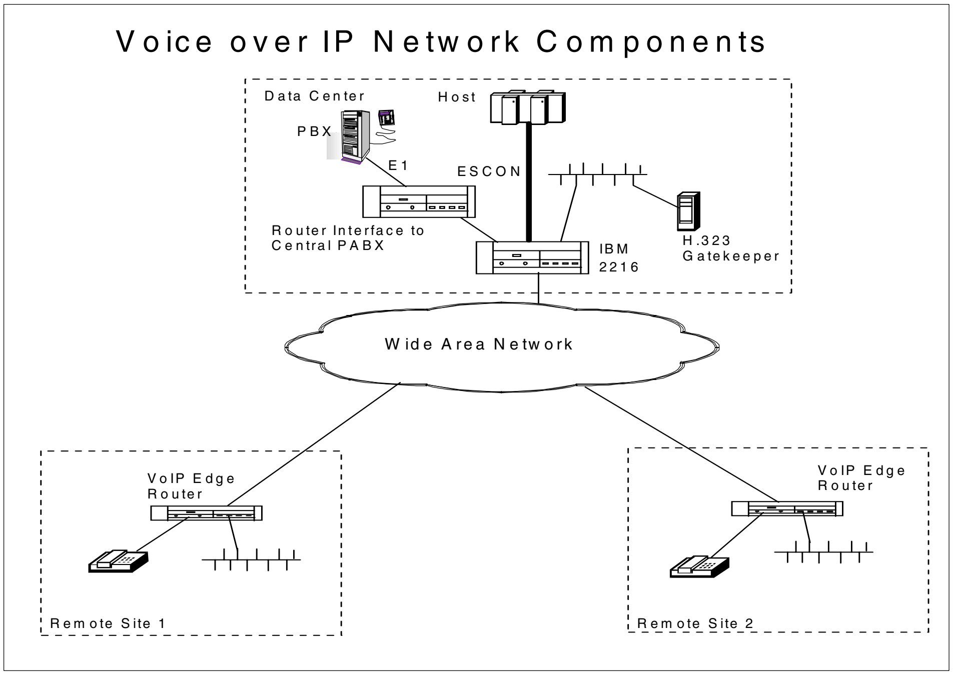


Figure 19. Voice over IP in a frame relay network environment

The principle components of this configuration are:

- Edge routers, which provide direct voice equipment interfaces and an H.323 Gateway function. These routers digitize and compress the voice traffic and transmit it through the WAN using the Real Time Protocol (RTP).
- Core network routers, which that simply route the IP voice packets and normal data through the WAN but do not provide any direct voice interfaces.
- Central Voice routers, which provide high-speed digital interfaces via E1 or T1 to large PABXs.
- H.323 Gatekeeper, which provides call control and telephone number to IP address mapping information. The H.323 Gatekeeper function is optional

and is not required if each edge device contains its own configuration information providing a mapping of telephone numbers to IP addresses. This approach is impractical in a large network as it would require significant manual reconfigurations in each edge router whenever telephone number changes occurred.

3.1.2 Configuring VoIP support on IBM and Cisco routers

A number of features need to be added to a basic IP data routing configuration for the successful transport of delay-sensitive voice traffic alongside IP data traffic. This section reviews these network features and shows how they are configured on both IBM and Cisco routers to support a mixed network environment. These features provide the following functions, which are not normally necessary in a data-only IP network:

- Fragmentation of large data frames and the interleaving of small voice packets among the fragments of data.
- Compression of the normally large IP packet headers associated with voice packets to achieve bandwidth efficiencies on low speed links.
- Prioritization of voice packets into special queues that always take priority over data packets.

3.1.2.1 Fragmentation and interleaving

To ensure that IP voice frames are transmitted at a constant and consistent rate and are not held up by large data frames requires a mechanism to fragment large data frames and interleave them with short voice frames. The recommended mechanisms for achieving this are:

- Multilink PPP (MLPPP) with interleaving on PPP links
- FRF.12 fragmentation on frame relay links

Fragmentation and interleaving on PPP links

Multilink PPP with interleaving is supported on both IBM and Cisco routers on ISDN basic rate and primary rate interfaces, on other dial-up links and on PPP serial links. One or more physical PPP interfaces are configured and assigned to a multilink PPP bundle. PPP interleaving is then enabled on the bundle. Note that even a single PPP link must be defined as a multilink PPP interface in order to allow interleaving to occur[1]. The multilink support on IBM and Cisco routers is compatible and configurations represented in Figure 24 on page 59 show the required parameters.

[1] PPP supports fragmentation but not interleaving, which is vital if we want to allow voice packet to “overtake” data packets.

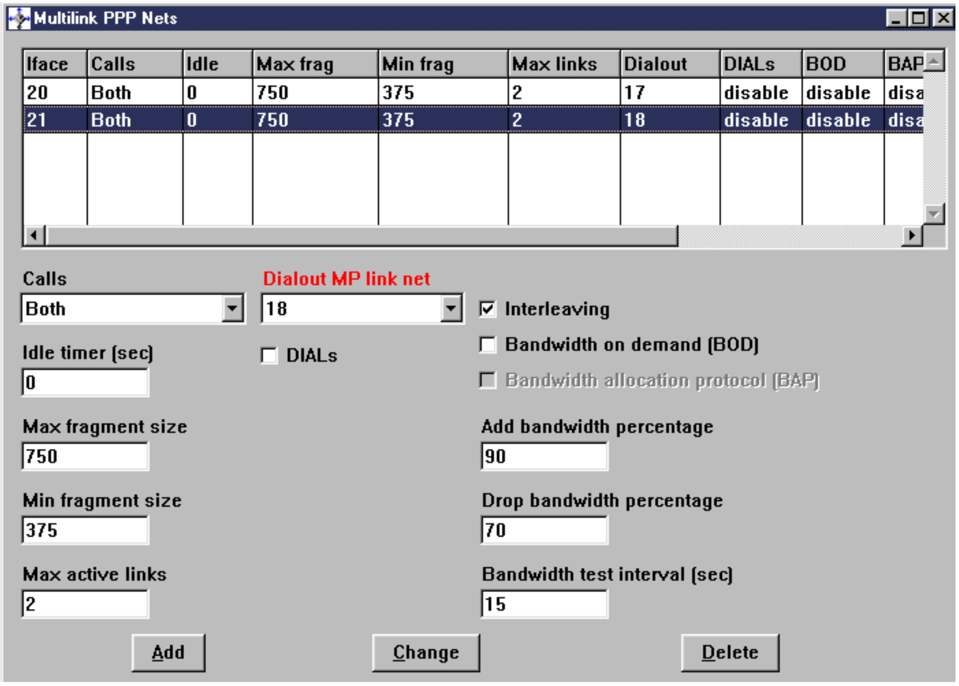
Interleaving and fragmentation on Cisco routers is configured with the following commands:

```
ppp multilink fragment-delay 20
```

```
ppp multilink interleave
```

The fragment delay parameter specifies that the maximum delay between voice fragments is 20 ms. The router will determine the appropriate fragment size based on this parameter and the link bandwidth.

IBM routers on the other hand require that actual fragment sizes be specified. You specifiy the maximum allowable packet size before fragmentation occurs, and the minimum fragment size that will be created by the process during definition of the MLPPP link as shown in Figure 20.



Iface	Calls	Idle	Max frag	Min frag	Max links	Dialout	DIALs	BOD	BAP
20	Both	0	750	375	2	17	disable	disable	disa
21	Both	0	750	375	2	18	disable	disable	disa

Calls: Both

Dialout MP link net: 18

☑ Interleaving

□ Bandwidth on demand (BOD)

□ Bandwidth allocation protocol (BAP)

Idle timer (sec): 0

□ DIALs

Max fragment size: 750

Min fragment size: 375

Max active links: 2

Add bandwidth percentage: 90

Drop bandwidth percentage: 70

Bandwidth test interval (sec): 15

Figure 20. Specification of interleaving and fragment sizes

This requires that an appropriate size be chosen based on the speed of the link to ensure similar 20 ms (or less) intervals between voice packets. The maximum packet size allowed should be large enough so that the largest RTP/UDP/IP voice packet is never fragmented; it is only larger data packets that are to be fragmented.

The IBM defaults are:

- 750 bytes - maximum frame size allowed without fragmentation
- 375 bytes - minimum fragment size created

These are too large for low-speed links. For example, a 750 byte data frame will take 90 ms to transmit on a 64 kbps link. A maximum fragment size of approximately 100 bytes will be necessary to keep voice frame delays comfortably below 20 ms.

The design aim should be to keep round-trip delay time for voice traffic across the entire network to 500 ms or less, which tends to be the point at which users start to complain. It's not easy to calculate what actual delay times will be in a network, not least because there are many causes of delay, starting with the delay inherent in the initial digitization process and in the transmission delay times over the physical links themselves. By setting the fragment size small enough, as we have done here, all we have done is to put an upper bound on one component of this delay in a single router in the network - we have said that we will not delay any single voice packet from waiting in the outbound transmission queue of the router for more than 20 ms This seems to be a reasonable starting point, but the only way to ensure acceptable round-trip delay times across the network is by considering the entire network and all components that make up the delay in aggregate, remembering that the round-trip delay time needs to be made up by adding the total delays in both directions across the network.

DIffering perspectives

There is obviously potential for confusion and mistakes when working with both IBM and Cisco VoIP parameters: Cisco takes the viewpoint that what is important is the maximum delay for voice packets, whereas IBM makes you work this out by actually making you configure the size of the largest data packet that can be sent on the link without fragmentation.

Watch out, though, that in the Cisco implementation the router is correctly configured with the speed of the serial line. If clocking is provided by the network, which is the normal configuration, the actual line speed may differ from the one configured in the router. If so, the actual voice delay may also differ from the value configured and expected. Ensure that the `bandwidth` command is correctly configured on the base serial interface (Serial0/0 in our examples).

Assuming the correct bandwidth parameter has been specified, the Cisco router calculates the maximum data fragment size for you. In some of our tests we specified 20 ms as the maximum delay (using the `ppp multilink fragment-delay 20` command). Given a bandwidth of 64 kbps, the number of bytes that will be transmitted in 20 milliseconds can be calculated as:

$$\frac{64 \times 1000 \times 20}{1000 \times 8} = 160$$

We then observed that large data packets were actually being transmitted as 156-byte fragments according to the MLP.001 ELS trace entry in the receiving IBM 2216, which was in fact 160 bytes including the PPP header or 152 bytes of actual data excluding MLPPP and PPP headers.

The reason for wanting to compare the actual fragment sizes used by IBM and Cisco is not that they are required to match, but simply that a network designer ought to want the actions of different devices to be consistent across a single network.

Fragmentation and interleaving on frame relay links

Frame relay links achieve similar fragmentation using FRF.12[2]. It is recommended by both IBM and Cisco that FRF.12 be used on frame relay links for both native VoFR traffic and for VoIP traffic in the frame relay environment.

[2] Frame Relay Fragmentation Implementation Agreement, FRF.12, The Frame Relay Forum, December 1997.

IBM and Cisco routers have compatible FRF.12 fragmentation implementations. IBM requires that FRF.12 be specified for the physical interface and then on each PVC where fragmentation is required. Cisco allows it to be specified only at the PVC level; however, both IBM and Cisco recommend that FRF.12 be applied to all PVCs on any physical link that is carrying voice traffic.

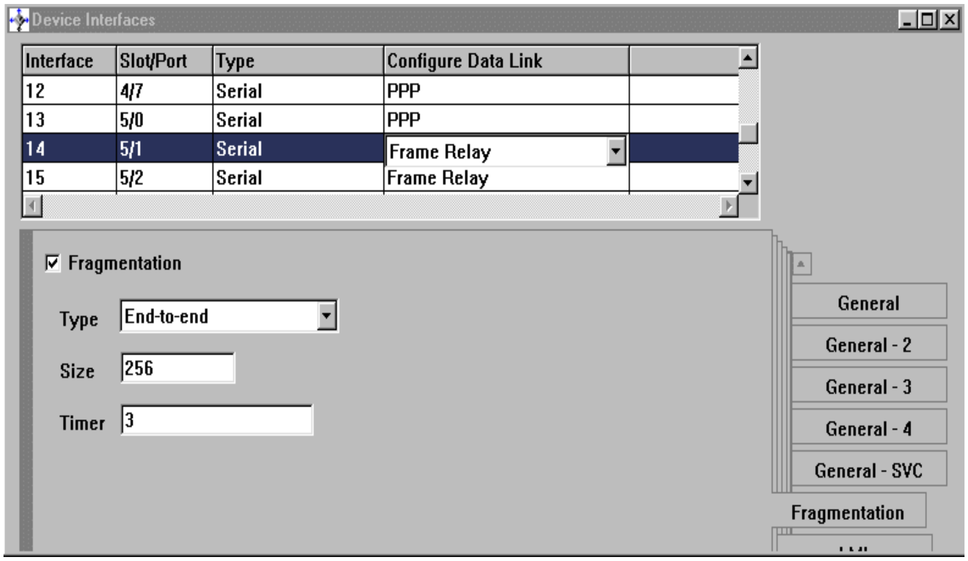
Note that FRF.12 specifies that end-to-end fragmentation is restricted to use on frame relay PVCs only. No current frame relay switch implements UNI/NNI fragmentation so although this is supported by IBM routers (although not by Cisco routers) it is extremely unlikely to be used in practice.

FRF.12 fragmentation is specified on Cisco routers using the following command.

```
frame-relay fragment 80
```

This example specifies a maximum fragment size of 80 bytes, which Cisco recommends on a 64 kbps link; if not specified, the default fragment size is 53 bytes. This size is the size of the payload in the fragment, excluding frame relay headers.

FRF.12 fragmentation is specified on IBM routers during frame relay link definition on the panel shown in Figure 21.



Interface	Slot/Port	Type	Configure Data Link
12	4/7	Serial	PPP
13	5/0	Serial	PPP
14	5/1	Serial	Frame Relay
15	5/2	Serial	Frame Relay

☑ Fragmentation

Type End-to-end

Size 256

Timer 3

Figure 21. Enable fragmentation on the IBM router frame relay interface

The fragment size for IBM routers can take any value in the range 50-8190 with a default value of 256.

Fragmentation must first be enabled here at the physical link level before it can be specified on any individual PVC. The default fragment type of UNI/NNI should normally be changed to End to End unless the frame relay service provider supports UNI/NNI fragmentation (which is extremely unlikely). The fragment size must be specified to suite the link speed, generally 100 bytes or less on a 64 kbps link. The “timer” parameter specifies how many seconds to wait for the next fragment in a sequence before discarding fragments already received.

Note - FRF.12 fragmentation

When using the IBM graphical configurator for IBM routers, If end-to-end fragmentation has been specified on IBM routers and fragmentation parameters have been set at the link level, it is also necessary to specifically enable fragmentation and set the fragment size for each PVC on the link. If you do not specify these parameters for each PVC then fragmentation will not occur on those PVCs and communication with Cisco routers using FRF.12 fragmentation will fail.

Older recommendations for minimizing data frame sizes often suggested reducing the size of large data frames by reducing the IP MTU size on the link. This forces all large data frames to be broken up into smaller IP packets. This creates additional processing overhead in the router, however, and can cause problems for some applications. The use of layer 2 techniques of MLPPP for PPP links and FRF.12 for frame relay links is now the preferred approach and is recommended by both IBM and Cisco. The Cisco Configmaker tool[3] (Version 2.4) generates IOS configurations for MLPPP that include MLPPP fragmentation and interleaving. However, the generated IOS configuration for a frame relay environment sets the MTU size to 240 bytes and does not use FRF.12. This is because FRF.12 only became available for most Cisco routers in the IOS code release 12.0(4)T. It is recommended that the configuration produced by Configmaker be changed to specify FRF.12 fragmentation instead, if possible.

3.1.2.2 RTP compression

Voice traffic is transported using the Real Time Protocol[4] (RTP), which means that each packet contains an IP header, a UDP header and an RTP header. H.323 terminals or gateways that transport voice traffic over low-speed WAN

[3] A graphical configuration tool available from the Cisco Web site (`http://www.cisco.com`), which can be used to generate configuration files for the small- and medium-sized ranges of Cisco routers.

[4] RFC 1889: RTP, A Transport Protocol for Real-Time Applications

links typically use G.729 or similar voice encoding schemes. G.729 typically results in a 20-byte voice packet payload that requires data transmission speeds of 8 kbps per voice channel; the RTP/UDP/IP headers add another 40 bytes to each packet. Without modification, G.729 VoIP streams each require 24 kbps bandwidth. This may be acceptable on high-speed networks where voice traffic is often carried at 64 kbps or 32 kbps today. However, it is usually necessary to reduce this overhead on low-speed links. RTP header compression[5] is usually employed to reduce the overhead of the RTP/UDP/IP packet header on these links to typically 5 bytes. This results in a bandwidth requirement of approximately 12 kbps for a single voice stream and is comparable to a G.729 compressed native voice over frame relay implementation.

IBM routers implement RTP compression for VoIP traffic on PPP links but not on frame relay links. RFC 2508 describes the RTP compression standard for PPP links. No standards-based approach exists for frame relay links at this time.

Cisco routers implement RTP compression for VoIP traffic on both PPP and frame relay links. The compression on PPP links is standards-based and conforms to RFC 2508 allowing compatible operation with IBM routers over MLPPP links. Cisco's RTP compression on frame relay is proprietary (since there is no standard) and also requires that the frame relay link uses Cisco's proprietary frame encapsulation. This means that IBM and Cisco routers should be able to fully interoperate in an MLPPP environment carrying voice and data traffic. However, in a frame relay environment compressed VoIP PVCs originating in a Cisco router cannot be interpreted by an IBM router and can only be switched through it using the frame relay frame handler FRFH support that became available in Version 3.4 of the IBM router common code.

[5] RFC 2507, IP Header Compression, RFC 2508, Compressing IP/UDP/RTP Headers for Low-Speed Serial Links and RFC 2509, IP Header Compression over PPP

Note

We could not implement RTP header compression over MLPPP between IBM and Cisco routers in our test environment. We took some traces and it appears there is an incompatibility in the MLPPP/LCP negotiation that establishes the link. The problem has been partially addressed by the fix for a Cisco bug (identifier CSCdp79896). This corrects the Cisco code to comply with RFC 2509 so that the link can be established. A compatibility problem still remains because IBM uses the source UDP port number to identify voice packets whereas Cisco uses the destination UDP port number; IBM is currently developing a code fix to address this particular issue. We have to admit, therefore, that our testing was performed without the use of RTP header compression, which is unrealistic in the sense that low-speed links really require it. Please bear this in mind when reading the rest of this chapter!

RTP header compression is implemented in Cisco routers using the following commands on either MLPPP or frame relay links:

```
ip rtp header-compression
```

```
ip rtp compression-connections number
```

The second statement is optional; by default the number of connections is 16, the same as the IBM default.

IBM routers only allow RTP compression to be enabled on PPP links as shown in Figure 22 on page 52. The `set ipcp` command also allows you to specify the UDP port numbers to use. By default Cisco allocate UDP ports 16384 to 32767 for RTP traffic and IBM allocates ports 5004 to 5515. In order to provide consistency throughout the network for compression and prioritization of VoIP traffic it is important that these values match in all routers. IBM routers allow you to easily change the UDP ports utilized for RTP traffic as shown in Figure 22 on page 52.

```
PPP 0 Config>set ipcp 1
IP COMPRESSION [yes]:
VJ or RTP Header Compression [RTP]:
Max Period: [256]?
Max Time: [5]?
Max Header: [168]?
RTP Start Port: [16384]? 2
RTP End Port: [16483]? 3 ***BUT SEE NOTE BELOW***
Number of TCP Slots: [16]?
Number of Non-TCP Slots: [16]?
Request an IP address [no]:
Send our IP address [no]:
Note: unnumbered interface addresses will not be sent.
Interface remote IP address to offer if requested (0.0.0.0 for none)
[0.0.0.0]?
```

Figure 22. IBM router RTP compression configuration

While it is possible to enable RTP header compression using the IBM router's graphical configuration tool, it is not currently possible to specify the start and end RTP port numbers using this tool; therefore, RTP compression configuration must currently be performed using the command line (`talk 6`) configuration process as shown in Figure 22 above.

The key points to note in the configuration are:

- Using command line configuration of an IBM router under `talk 6` the IPCP parameters are set on the MLPPP interface. 1
- The starting UDP port number for RTP voice traffic defaults to port 5004 on IBM routers. Cisco's implementation is set to a starting port of 16384 and cannot be altered, so it is necessary to also specify port 16384 as the starting port number on IBM configurations. 2
- The end UDP port number for RTP voice traffic is specified. IBM defaults to port 5515. The Cisco implementation specifies a number of ports beyond 16384 with a default of 16383 ports. To be consistent with the Cisco implementation the ending port number on IBM routers should be set to 32767 (16383 port range). 3

We got it wrong

As should be apparent from Figure 22 on page 52, we originally believed that Cisco's implementation only uses a range of 100 port numbers, from 16384 to 16483. We were wrong. Since we feel that this book should serve to document our experiences, rather than hide our embarrassment, we wanted to leave this mistake in the book as a warning to our readers. Our subsequent experience confirmed that Cisco routers generate voice traffic using UDP port numbers significantly higher than 16483.

3.1.2.3 Prioritization of VoIP traffic in IBM and Cisco routers

Both IBM and Cisco offer comprehensive bandwidth allocation and prioritization schemes to manage router traffic. In this section we review the specific elements of these schemes that are necessary to provide a compatible VoIP implementation.

IBM offers:

- Bandwidth Reservation System (BRS), which allows bandwidth allocation and prioritization for IP as well as other multiprotocol traffic. In particular BRS now provides a Super Class for RTP voice traffic. Packets in the Super Class will always be transmitted before any other packets.
- Differentiated Services (DiffServ), which provides a standards-based approach for implementing bandwidth allocation and prioritization for IP traffic. In particular, DiffServ implements an Expedited Forwarding (EF) premium queue for traffic such as RTP voice traffic. Packets in the EF queue are transmitted before packets in any other queue, up to specified bandwidth limits.

BRS and DiffServ are mutually exclusive; only one can be enabled on any particular interface.

The key differences are:

- DiffServ allows you to implement a strict upper limit to bandwidth available to a particular protocol, such as VoIP traffic. Frames in excess of this limit will be dropped, whereas BRS guarantees minimum bandwidth available to a particular class of traffic but will not limit the bandwidth available to a certain protocol. In particular, packets in the Super Class will always be transmitted first even if they use all the available bandwidth. DiffServ will allow you to limit the amount of voice traffic that any router can inject into the IP network. DiffServ will discard EF class packets in excess of the specified bandwidth.

- DiffServ is an IP standard and is not effective for allocating bandwidth to other protocols such as SNA and NetBIOS. If multiple protocols need to be managed then BRS is the only IBM option.

Cisco offers a great range of prioritization and queueing options, some of which are limited to particular hardware platforms. In general the two schemes that appear to be most commonly used with the Cisco 2600 and 3600 series routes are:

- Weighted Fair Queuing (WFQ) - WFQ is enabled by default on interfaces with bandwidth less than 2 Mbps when protocols such as IP and IPX are being used. It is simple to configure and automatically assigns traffic to queues to ensure that all traffic receives sufficient bandwidth. By itself WFQ is not appropriate for voice traffic, since it does not provide guaranteed automatic prioritization of voice traffic. However, it does provide an intrinsic absolute priority queue to which you can assign all RTP voice traffic by the configuration command `ip rtp priority`. WFQ is also referred to as “fair-queuing” in Cisco documentation.
- Class Based Weighted Fair Queuing (CBWFQ) - This technique allows specific percentages of bandwidth to be allocated for classes of traffic and is useful in an environment that includes SNA traffic, since it allows you to configure a specific percentage of bandwidth for SNA. This technique has only become available in more recent releases of IOS and is recommended by Cisco in preference to older techniques, such as Custom Queuing. CBWFQ provides more classification and control over traffic than simple WFQ, but requires more manual configuration. CBWFQ also allocates the special absolute priority queue that can be utilized via the `ip rtp priority` command.

The Cisco `ip rtp priority` option can be applied to either the WFQ or CBWFQ environment. Within a WFQ system the statement has the following form:

```
ip rtp priority (starting rtp port number) (range of ports) (bandwidth)
```

The starting UDP port number and range of ports specify the UDP ports that RTP VoIP traffic will use on the router. Any traffic using these ports will be placed in the strict priority queue and be forwarded before all other traffic, but only up to the bandwidth limit specified. RTP packets in excess of this bandwidth will be dropped. In this way the `ip rtp priority` feature operates in a fashion similar to the DiffServ EF queue within IBM routers.

Note that the bandwidth parameter is specified in terms of actual VoIP compressed traffic, if RTP header compression is in use. Typically a G.729 VoIP data stream before RTP compression is applied will require approximately 24 kbps, but after compression is likely to require only 12 kbps. Therefore it is only necessary to allocate 12 kbps bandwidth for each concurrent voice call that you want to support.

It is also recommended that all VoIP traffic be tagged as critical traffic through use of the precedence bits in the IP type of service (TOS) field. This traffic needs to be tagged by the originating router. For example the following Cisco configuration statements will tag all VoIP traffic destined for a particular dial peer with an IP precedence setting of “5”, which is the highest priority possible for user data traffic.

```
dial-peer voice 3 voip
 codec g729r8
 ip precedence 5
 session target ipv4:10.1.100.2
 vad
 destination-pattern 321321
```

Precedence and type of service

Here we have an issue between the terminology used by Cisco and IBM. Cisco is using the definitions of RFC 1349, which defines the first three bits in the “service type” byte of the IP header (the second byte of the IPv4 header, in fact) to indicate “precedence”. RFC1349 also assigns names to each of the eight different precedence values, ranging from “routine” to “critical” and “network” precedence settings, and the related Cisco command `set ip precedence` even allows the use of these terms instead of a numeric value. But here, with the `ip precedence` command, only a numeric value between 0 and 7 is allowed, which directly maps to the binary representation of the first three bits in the “service type” byte. Incidentally, there is also an `ip tos` command that allows the next four bits in the byte to be set by providing a numeric value between 0 and 15.

The problem with RFC 1349 is that it has been superseded and that the definition of the “service type” byte has changed and will probably change again in the future (see RFC 2474, for example). IBM routers have chosen to treat the complete “service type” byte as a single entity, and allow modification or observation of any or all of the bits in the byte, even though RFC 2474 currently defines the last two bits as “currently unused”.

What this means is that a Cisco definition of `ip precedence 5` equates to an IBM interpretation in which the first three bits in the byte are examined for the binary value 101. In IBM terms, this means applying (a logical AND) a “mask” of 11100000 to the byte and then looking for value of 10100000. In hexadecimal, the value of the mask is E0 and the value to match is A0.

This tag is not used by Cisco routers along the traffic path that support the `ip rtp priority` feature, since these routers will select the VoIP traffic by UDP port, and will place the voice traffic in the special guaranteed priority queue. However, not all Cisco routers support this feature and these routers will rely on the IP Precedence setting to allocate priority treatment to the VoIP traffic.

A consistent prioritization and control system needs to be implemented in a mixed network of IBM and Cisco routers. It is likely that all VoIP traffic will be introduced into the network via Cisco routers with voice interfaces such as FXS or E&M interfaces. If the network is a relatively simple IP data network then it is likely that WFQ with the use of the `ip rtp priority` feature on the Cisco routers will provide the necessary voice prioritization. If the network includes SNA or other bandwidth-sensitive traffic, then it is preferable to implement CBWFQ on the Cisco routers to ensure sufficient SNA bandwidth even if the SNA traffic is carried via DLSw.

The issue of consistency is that all routers in the network need to be able to recognize VoIP traffic and treat it accordingly; furthermore, traffic that isn't voice traffic should not masquerade as voice traffic in order to receive preferential treatment. Voice traffic can be identified by the UDP port number falling in a specified range, but the Cisco `ip priority` command also allows each voice packet to be marked in a particular manner. If the network is under control to the extent that *only* voice packets have a specific priority/tos setting (to use the Cisco terminology), all network routers can identify and classify voice packets by use of the "service type" byte in the IP header.

IBM routers in the core of the network need to use either DiffServ or BRS to provide the necessary voice prioritization and bandwidth control. If the network contains IP only, and it is important to put an upper limit on voice traffic, then DiffServ is the best choice, since DiffServ provides a strict bandwidth limit on voice traffic in a fashion very similar to the Cisco `ip rtp priority` feature. DiffServ also provides an Assured Forwarding queue designed for traffic such as SNA data (encapsulated in IP) that requires a specific transmission rate but can tolerate slight delay variations. If the network contains native APPN traffic, IPX, or bridged traffic, then BRS is the only choice. BRS is also much simpler to configure.

Another important point is that DiffServ is not supported on frame relay interfaces configured for FRF.12 fragmentation on an IBM router. As most frame relay links carrying voice traffic (below T1 speeds) will use FRF.12 fragmentation, then BRS is the only option for these links.

3.1.3 Voice over IP configuration example using MLPPP

We now review the configuration parameters required to implement a mixed IBM and Cisco router VoIP network using PPP data links as shown in Figure 23.

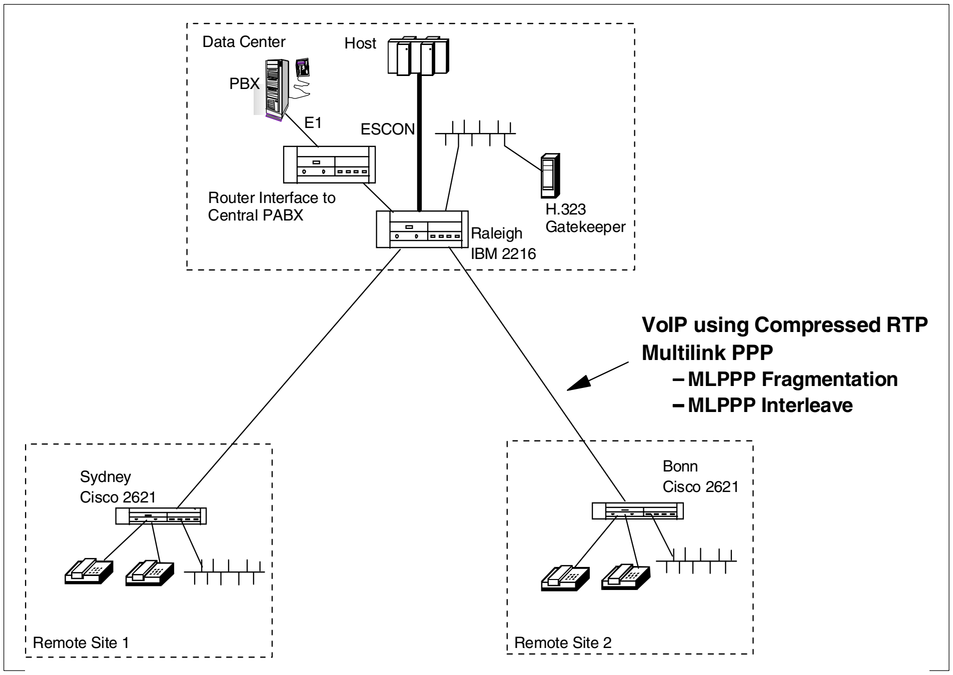


Figure 23. VoIP configuration using PPP links - mixed IBM and Cisco environment

Configuring the Cisco remote site routers

The configuration statements for the Sydney-Cisco 2621 required to implement the link efficiency and prioritization schemes described in the previous sections are shown in Figure 24 on page 59. This is not the complete Cisco router configuration, since we are specifically addressing parameters relating for VoIP transport here. Complete Cisco router configurations supporting the underlying IP routing environment are reviewed in Chapter 2, “Dynamic routing protocols” on page 13.

```
interface Serial 0/0
 no shutdown
 description connected to IBM_2216
 bandwidth 64
 clock rate 64000
 no ip address
 encapsulation ppp
 ppp multilink 1
!
interface Virtual-Template 1 2
 description connected to IBM_2216
 ip unnumbered Loopback 0 3
 encapsulation ppp
 ip rtp header-compression 4
 ip rtp priority 16384 100 24 5
 ppp multilink 6
 ppp multilink interleave 7
 ppp multilink fragment-delay 30 8
 fair-queue 64 256 0 9
!
multilink virtual-template 1 10
!
interface Loopback0
 ip address 10.1.100.2 255.255.255.0
!
voice-port 1/0/0 11
 no shutdown
 description connected to 321311 (321321)
 comfort-noise
 cptone US
 signal loopstart
!
dial-peer voice 1 pots 12
 port 1/0/0
 destination-pattern 321321 13
!
dial-peer voice 2 voip 14
 codec g729r8 15
 ip precedence 5 16
 session target ipv4:10.1.102.2 17
 vad
 destination-pattern 321311
!
! VoIP phone number-to-extension database
!
num-exp 1.. 3213.. 18
```

Figure 24. Sydney Cisco 2621 VoIP over MLPPP configuration

The key points to note are:

- The serial interface 0/0 is assigned to the PPP multilink bundle. 1

- A Virtual Template is created to define the characteristics of the multilink PPP bundle. 2
- While the MLPPP interface is unnumbered it uses the Loopback IP address as the source IP address for packets it transmits. 3
- RTP header compression is enabled with the default 16 slots (remember that we didn’t actually get this to work ourselves). 4
- The `ip rtp priority` feature is enabled to ensure strict priority for voice packets. In this case up to 24 kbps of uncompressed VoIP bandwidth is allocated for voice, equivalent to two voice calls at a time. The UDP ports specified for RTP are 16384 and the following 100 ports (remember that this is incorrect and should have a larger port number range configured). 5
- Multilink PPP with fragmentation and interleaving with a maximum delay between packets of 30 ms is specified. 6 7 and 8
- WFQ is enabled on the interface, which implicitly creates the strict priority queue used by RTP traffic. 9
- The Virtual Template is applied to the Multilink PPP bundle - in this case one 64 kbps link. 10
- An FXS voice port on the Cisco 2621 is configured. 11
- A link between the FXS voice port and a plain old telephone handset (POTS) and its associated full phone number is created 12 and 13
- A connection to a remote peer using VoIP is created. 14
- The G.729 compression algorithm is specified to create a voice data stream using approximately 8 kbps. 15
- IP precedence of “5” for critical traffic is assigned to all VoIP packets sent over this link. 16
- The target IP address associated with the target telephone number is specified. 17
- A short telephone number calling plan is defined in which an initial dialled number “1” equates to “3213”. So, in order to call “321321” we only need to dial “121”. 18

The Bonn-Cisco 2621 router shown in Figure 23 on page 58 is configured in the same manner.

Configuring the IBM 2216 central site router

The IBM 2216 must also be configured to support:

- MLPPP fragmentation and interleaving

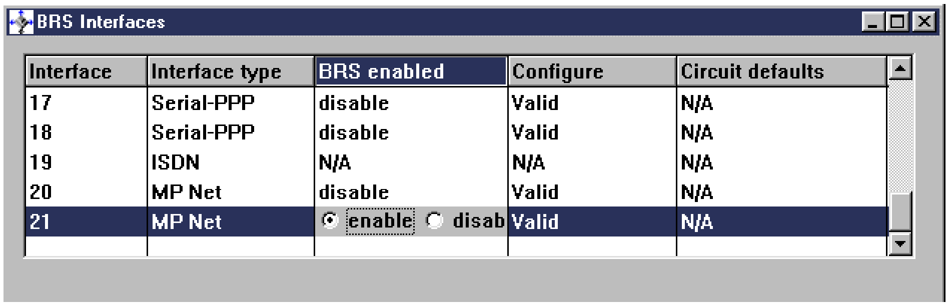
- RTP Header Compression
- Bandwidth management using either BRS or DiffServ

We assume that the reader is familiar with configuring standard MLPPP support on IBM routers, since this is often required to provide aggregation of bandwidth on multiple ISDN ‘B’ channels. In this case, however, MLPPP must be specified even if a single physical PPP link is utilized, since this is the only way to achieve Interleaving between IP voice packets and data fragments. Once the basic MLPPP configuration is complete it is necessary to configure parameters to specify fragmentation and to enable interleaving as shown in Figure 20 on page 45. In this case the maximum fragment size should be set to approximately 100 bytes, since the link speed is only 64 kbps, and the minimum fragment size should be set to approximately 80 bytes. These setting will fragment large data frames but will ensure that uncompressed RTP/UPD/IP traffic is never fragmented.

RTP header compression is enabled as shown in Figure 22 on page 52. Again the UPD start port should be set to 16384 and the end port to 32767 to match the Cisco implementation.

In this configuration example we used BRS to prioritize voice traffic that needed to be carried through the IBM central 2216 router. The BRS configuration steps are reviewed below, including the creation of a Super Queue for voice traffic. We could alternatively have chosen to use DiffServ to prioritize voice traffic on these interfaces.

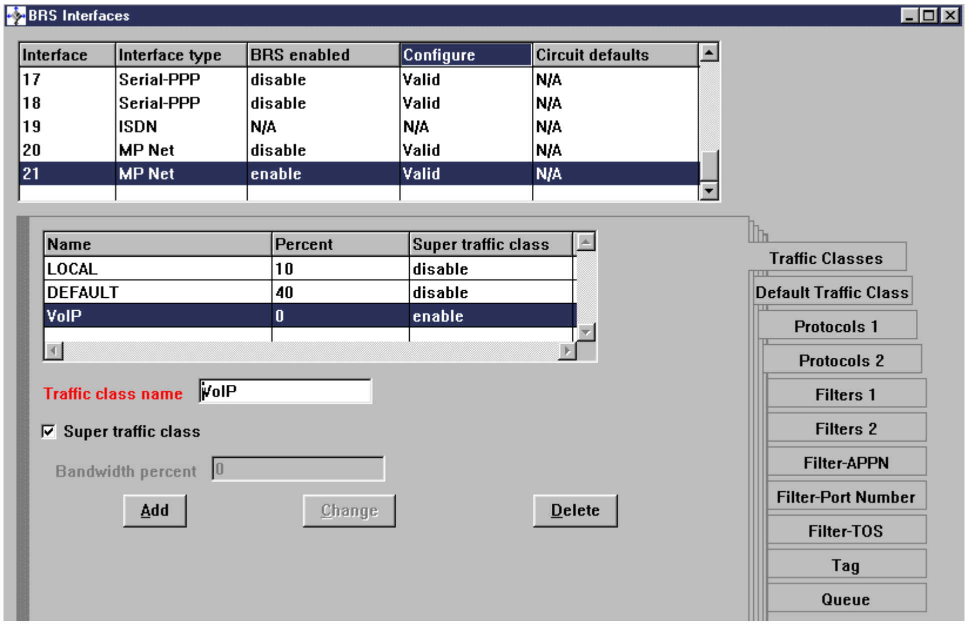
First we enable BRS on the Multilink PPP interface as shown in Figure 25.



Interface	Interface type	BRS enabled	Configure	Circuit defaults
17	Serial-PPP	disable	Valid	N/A
18	Serial-PPP	disable	Valid	N/A
19	ISDN	N/A	N/A	N/A
20	MP Net	disable	Valid	N/A
21	MP Net	● enable ○ disab	Valid	N/A

Figure 25. Enable BRS on the multilink PPP interface

Next we create a new BRS traffic class called VoIP and allocate it to the Super Traffic Class as shown in Figure 26.



Interface	Interface type	BRS enabled	Configure	Circuit defaults
17	Serial-PPP	disable	Valid	N/A
18	Serial-PPP	disable	Valid	N/A
19	ISDN	N/A	N/A	N/A
20	MP Net	disable	Valid	N/A
21	MP Net	enable	Valid	N/A

Name	Percent	Super traffic class
LOCAL	10	disable
DEFAULT	40	disable
VoIP	0	enable

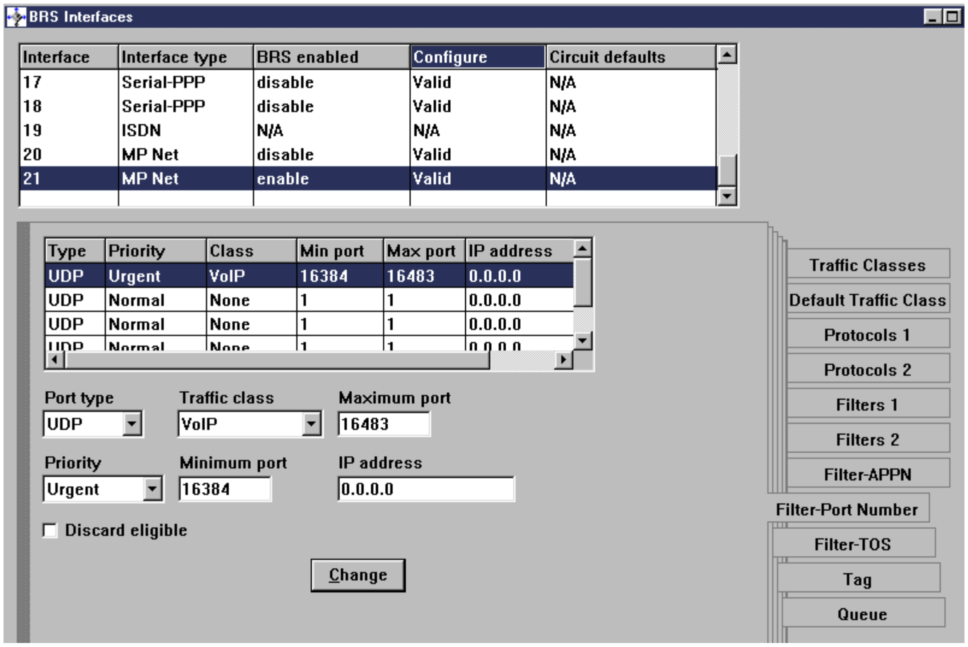
Traffic class name VoIP

☑ Super traffic class

Bandwidth percent 0

Figure 26. Creating VoIP traffic class as super traffic class

In the next step we assign the appropriate RTP VoIP traffic to the Super Class as shown in Figure 27 and Figure 28 on page 63. It is recommended that filtering of this traffic for assignment to the Super Class be performed by both UDP port number as shown in Figure 27 or by the IP precedence setting in the TOS byte as shown in Figure 28. Our mistake was in the range of port numbers used by Cisco routers originating voice traffic; the maximum port number should in fact have been set to 32767 here.



Interface	Interface type	BRS enabled	Configure	Circuit defaults
17	Serial-PPP	disable	Valid	N/A
18	Serial-PPP	disable	Valid	N/A
19	ISDN	N/A	N/A	N/A
20	MP Net	disable	Valid	N/A
21	MP Net	enable	Valid	N/A

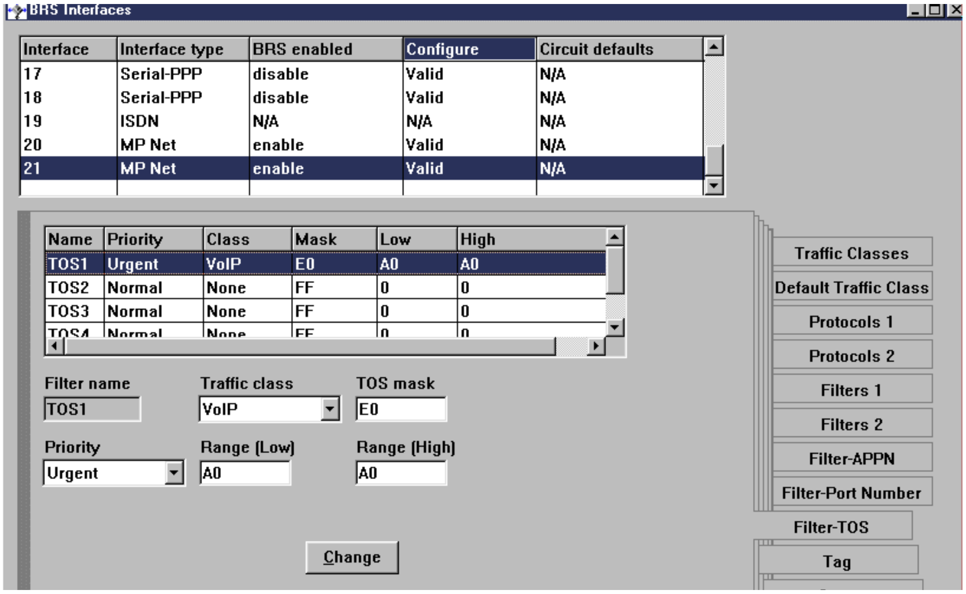
Type	Priority	Class	Min port	Max port	IP address
UDP	Urgent	VoIP	16384	16483	0.0.0.0
UDP	Normal	None	1	1	0.0.0.0
UDP	Normal	None	1	1	0.0.0.0
UDP	Normal	None	1	1	0.0.0.0

Port type: UDP
Traffic class: VoIP
Maximum port: 16483

Priority: Urgent
Minimum port: 16384
IP address: 0.0.0.0

☐ Discard eligible

Figure 27. Assigning VoIP traffic to the BRS super class based on UDP port number



Interface	Interface type	BRS enabled	Configure	Circuit defaults
17	Serial-PPP	disable	Valid	N/A
18	Serial-PPP	disable	Valid	N/A
19	ISDN	N/A	N/A	N/A
20	MP Net	enable	Valid	N/A
21	MP Net	enable	Valid	N/A

Name	Priority	Class	Mask	Low	High
TOS1	Urgent	VoIP	E0	A0	A0
TOS2	Normal	None	FF	0	0
TOS3	Normal	None	FF	0	0
TOS4	Normal	None	FF	0	0

Filter name: TOS1
Traffic class: VoIP
TOS mask: E0

Priority: Urgent
Range (Low): A0
Range (High): A0

Figure 28. Assign VoIP traffic to the super class based on IP precedence

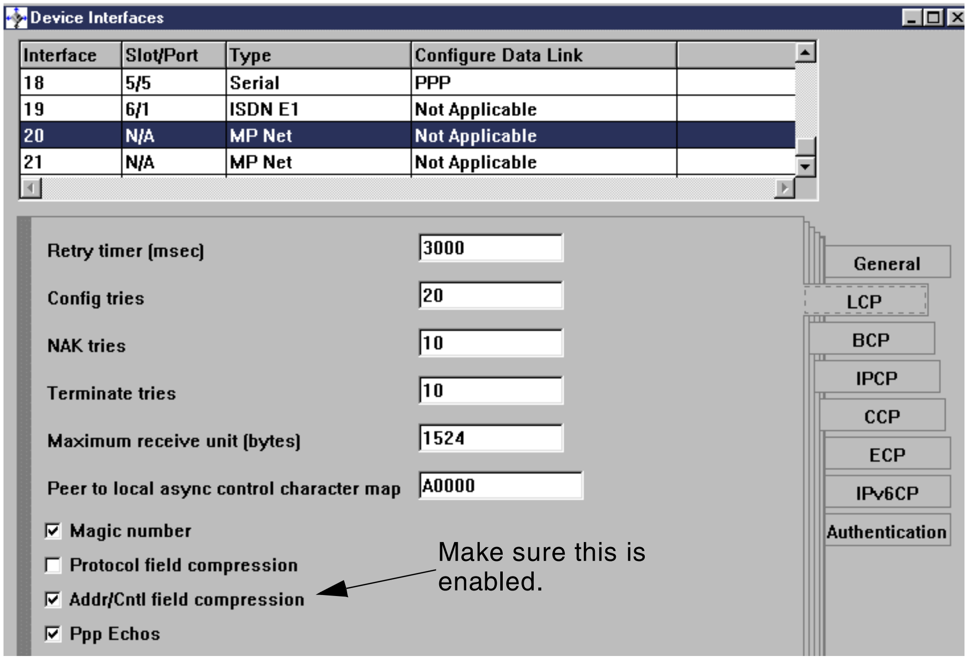
The BRS configuration shown in Figure 28 is specified to filter on only the first 3 bits of the TOS byte (Mask E0) and to select only traffic that has a precedence setting of 5 within the first three bits of the TOS byte (A0).

These steps assign all VoIP traffic to a queue that is always serviced before all other queues in a fashion similar to the Cisco `ip rtp priority` feature. However, note that BRS places no upper limit on the bandwidth that this voice traffic can use.

An additional parameter

In order to achieve compatible MLPPP fragmentation and interleaving of voice and data packets between the IBM and Cisco routers in this configuration example we needed to enable PPP Address Field/Control Field compression on the MLPPP link definitions on the IBM router. These are enabled on the LCP interface configuration panel for the MLPPP links as shown in Figure 29 on page 65.

It may be that future code changes by both IBM and Cisco mean that this parameter no longer needs to be specified. One thought we had is that perhaps once RTP header compression is working between IBM and Cisco then this parameter will no longer be required. There is no doubt, however, than in our configuration this parameter is required to allow voice and data traffic to flow across the network at the same time.



Interface	Slot/Port	Type	Configure Data Link
18	5/5	Serial	PPP
19	6/1	ISDN E1	Not Applicable
20	N/A	MP Net	Not Applicable
21	N/A	MP Net	Not Applicable

Retry timer (msec) 3000

Config tries 20

NAK tries 10

Terminate tries 10

Maximum receive unit (bytes) 1524

Peer to local async control character map A0000

☑ Magic number

☐ Protocol field compression

☑ Addr/Cntl field compression

☑ Ppp Echos

Make sure this is enabled.

Figure 29. Enable Address/Control field compression to achieve MLPPP compatibility

If this parameter was not enabled we observed[6] that when the Cisco router transmitted voice packets interleaved with data packets, the voice packets were considered to be “bad packets” by the IBM router and hence discarded. This resulted in very bad voice quality when voice and data were mixed on the links; the Cisco router was correctly interleaving voice packets between data packets but all the voice packets were then being thrown away by the IBM router. When there was no data traffic on the link, voice traffic alone was not discarded. Enabling this feature circumvented the problem (the IBM router no longer discarded voice packets) and resulted in high-quality voice transmission.

3.1.4 VoIP interoperability in the frame relay environment

A number of specific considerations apply when transporting VoIP traffic through a frame relay network. The key differences in the frame relay environment (when compared with PPP) relate to how RTP header compression is supported and how to cope with QoS required by voice traffic in a frame relay network. Remember that here we are still considering VoIP traffic carried on frame relay links, not native VoFR traffic. Native VoFR implementations are reviewed in 3.2, “Voice over frame relay” on page 78.

[6] Using the Event Logging System on the IBM router

No standards have been defined to support RTP header compression over frame relay links. Some vendors, including Cisco, have implemented proprietary RTP compression implementations for frame relay links; IBM routers do not support these. Therefore IBM routers do not support RTP header compression for VoIP traffic over frame relay links.

In a mixed IBM and Cisco network there are three options:

- Disable RTP header compression on all routers. This allows full interoperability for VoIP transport over frame relay, but is not very efficient on low-speed links.
- Enable RTP header compression on all Cisco routers and switch all traffic through the IBM routers via the FRFH function of the IBM routers.
- Use multiple frame relay PVCs between IBM and Cisco routers.
  - On one PVC, enable Cisco router RTP header compression for voice transport and switch all traffic for these PVCs through the IBM routers using the FRFH function.
  - On the other PVC, disable Cisco router RTP header compression and use RFC1490/IETF encapsulation for data traffic, which allows full routing interoperability between Cisco and IBM routers for these data PVCs.

These configuration options are shown in more detail in the following sections.

The other key consideration is that public frame relay networks offer no guarantees of QoS. It is therefore critical that we configure all routers to minimize the chance of voice packet loss in the network and make best use of the available frame relay CIR on each link, because if we send traffic in excess of the guaranteed CIR we run the risk of losing packets, and these lost packets could be voice packets. If we ensure we comply with CIR then - in theory - no packets will be dropped.

This is achieved by:

- Enforcing traffic shaping on frame relay interfaces of both IBM and Cisco routers.
- Adjusting FRF.12 fragment sizes
- Setting the appropriate frame relay committed burst size (B_c) for each interface and PVC.

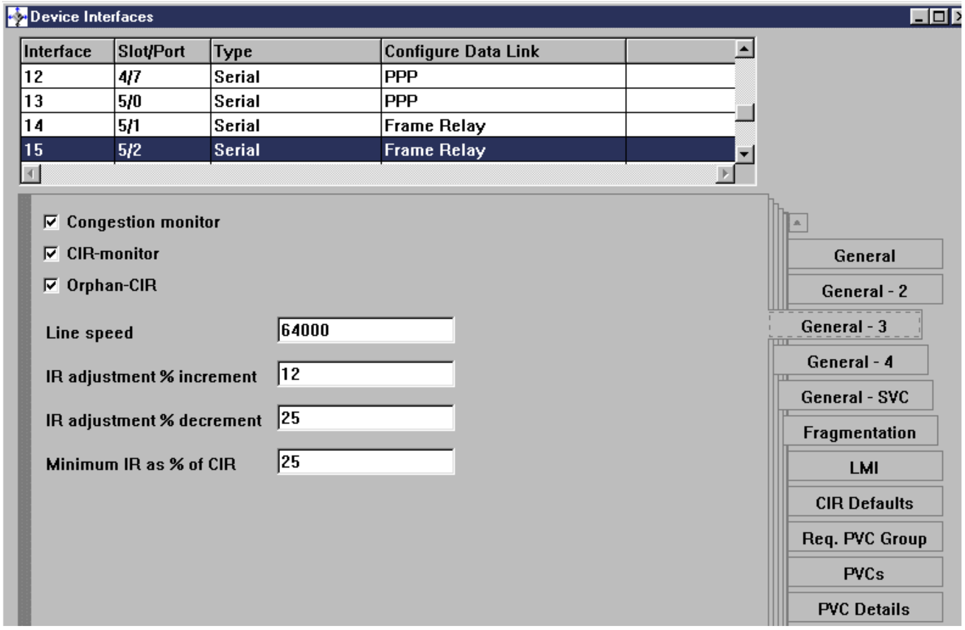
Combining voice and data traffic

The last consideration sounds so simple but has significant implications: one of the key differences between voice and data traffic is that data traffic can be retransmitted if it gets lost by the network. Existing data-only networks that use frame relay links may deliberately attempt to send more traffic over the frame relay links than the committed information rate (CIR); most of these packets probably make it through the network but those that don't can be retransmitted.

Once voice traffic is added to the network, it is probably going to be necessary to constrain devices connected to frame relay networks to comply with the CIR. This may then reduce the effective throughput of data traffic that may, in turn, lead to a requirement to pay more money to a frame relay service provider to increase the CIR on the link. This is one of the many considerations to be aware of when attempting to combine voice and data networks.

3.1.4.1 Setting frame relay parameters for VoIP on IBM routers

To enforce traffic shaping, enable the CIR monitor as shown in Figure 30.



Interface	Slot/Port	Type	Configure Data Link	
12	4/7	Serial	PPP	
13	5/0	Serial	PPP	
14	5/1	Serial	Frame Relay	
15	5/2	Serial	Frame Relay	

☑ Congestion monitor

☑ CIR-monitor

☑ Orphan-CIR

Line speed: 64000

IR adjustment % increment: 12

IR adjustment % decrement: 25

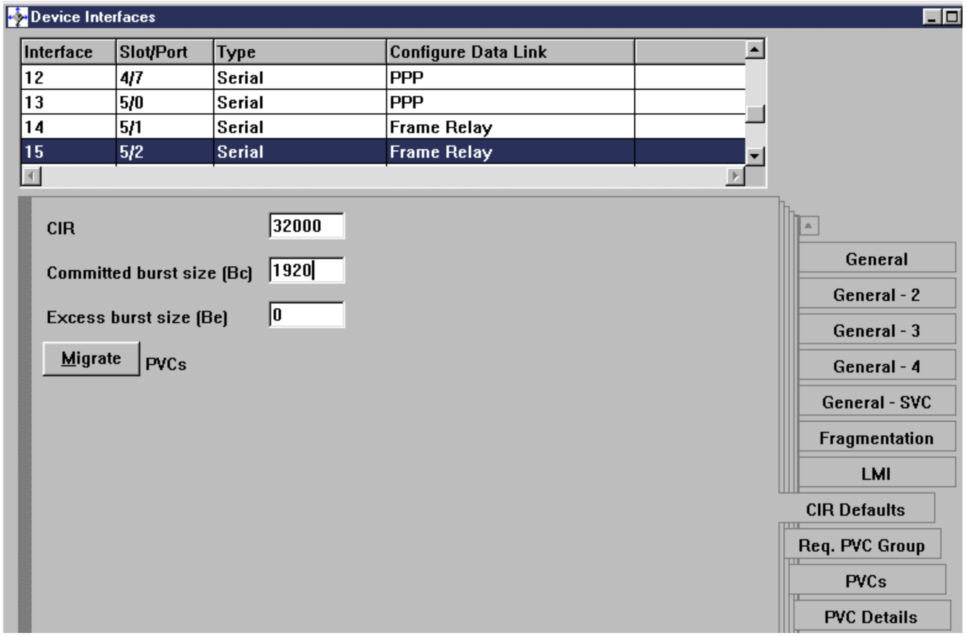
Minimum IR as % of CIR: 25

Figure 30. Enable CIR monitor to enforce traffic shaping

Enabling the CIR monitor overrides the congestion monitor if both are enabled.

Next we enable fragmentation on the interface and set the appropriate fragment size as shown in Figure 21 on page 48.

Finally the CIR and B_c values are set as shown in Figure 31.



Interface	Slot/Port	Type	Configure Data Link	
12	4/7	Serial	PPP	
13	5/0	Serial	PPP	
14	5/1	Serial	Frame Relay	
15	5/2	Serial	Frame Relay	

CIR 32000

Committed burst size (Bc) 1920

Excess burst size (Be) 0

Migrate PVCs

Figure 31. Set CIR and B_c values for the interface

3.1.4.2 Setting frame relay parameters for VoIP on Cisco routers

Similar frame relay traffic shaping parameters can be applied to Cisco routers. These parameters are shown in the sample configuration shown in Figure 32.

```
interface Serial0/0
no ip address
no ip directed-broadcast
encapsulation frame-relay 1
no ip mroute-cache
ip rtp header-compression iphc-format
frame-relay traffic-shaping 2
frame-relay lmi-type ansi
ip rtp priority 16384 100 24 3
!
interface Serial0/0.2 point-to-point
no ip directed-broadcast
frame-relay interface-dlci 172
class VoicePVC 4
!
map-class frame-relay VoicePVC 5
frame-relay cir 32000 6
frame-relay bc 1920 7
frame-relay be 0
frame-relay mincir 16000 8
frame-relay fair-queue 9
frame-relay fragment 100 10
```

Figure 32. Sample Cisco VoIP frame relay traffic shaping configuration

The key parameters here are:

- Cisco proprietary frame relay encapsulation is required for RTP compression over frame relay. 1
- Frame Relay Traffic Shaping (FRTS) is applied to the physical interface. Cisco FRTS is the mechanism that enforces the CIR and other frame relay settings. 2
- RTP traffic is assigned to the absolute priority queue within Weighted Fair Queuing. 3
- The frame relay traffic class we called VoicePVC is applied to this frame relay PVC. 4
- A frame relay map class is created to specify a set of frame relay traffic-shaping characteristics. 5
- The CIR is set and enforced by the FRTS system. 6

- The committed burst size is set to produce small traffic bursts with shorts delays between them. 7
- A frame relay minimum CIR can be set. This is a CIR that is enforced if link congestion occurs. 8
- WFQ is applied to set up the strict priority queue required for voice traffic. 9
- FRF.12 fragmentation is applied to the map class. 10

Note that the RTP compression setting requires Cisco proprietary frame relay encapsulation so this PVC (carrying VoIP traffic) cannot be terminated in an IBM router, but must be switched through it to a compatible Cisco router.

3.1.4.3 General considerations for setting frame relay parameters

The traffic characteristic we are aiming for is to achieve a steady stream of voice and FRF.12 data fragments with very small delays between each transmission. The rule governing transmissions on frame relay links is:

$B_c = CIR \; x \; T_c$

Where:

- B_c is the committed burst size in bits, the number of bits the network commits to deliver in the interval T_c seconds
- T_c is the burst interval in seconds
- CIR is the committed information rate in bits per second, the average rate at which the network commits to deliver traffic.

By default, IBM routers set both B_c and CIR to 64000, which therefore also means that T_c is set to 1. If the router is observing CIR, it could send 64000 bits in under a second and then have to wait for the start of another second before sending any more traffic. This interval is far too great for voice traffic[7]. We need the maximum interval between voice frames to be very small: between about 30ms to 60ms. If we aim for a maximum interframe gap of 30ms (T_c = 30/1000) we need to set B_c to 1920 if the committed information rate remains at 64000 bps; we need to set B_c to 960 if the committed information rate is 32 kbps as in our test network.

These B_c values equate to a committed burst size of packets of 240 bytes and 120 bytes respectively in the time interval of 30 ms. This means that the FRF.12 fragment size must be set considerably smaller than this value, and the exact fragment size needs to be calculated from the characteristics and number of simultaneous voice calls we want to permit over the link:

[7] If the access rate were 2 Mbps, this wait time could be as long as 31/32 seconds, or 0.97 seconds.

essentially the data fragment size has to be set to a small enough size to allow some data to be transmitted along with all the required voice packets in a single time interval. For a more detailed review of how to calculate appropriate fragment sizes, see Appendix B, “Sample calculations for frame relay parameters” on page 163.

Once these parameters have been set it is necessary to enforce them by enabling CIR monitor on the IBM routers and by applying the frame relay map class to the PVC in Cisco routers.

3.1.4.4 VoIP example using a single frame relay PVC

In order to implement a VoIP solution *between* IBM and Cisco routers using frame relay links, it is necessary to disable RTP header compression on all routers. This provides frame relay and IP routing compatibility between IBM and Cisco routers. This option may only be suitable for higher speed networks that do not require significant voice traffic compression. Only one DLCI is required for each remote site, since all traffic (data and voice) can be encapsulated in RFC1490/IETF frames; the traffic is then compatible with IBM and Cisco routers. This option is shown in Figure 33.

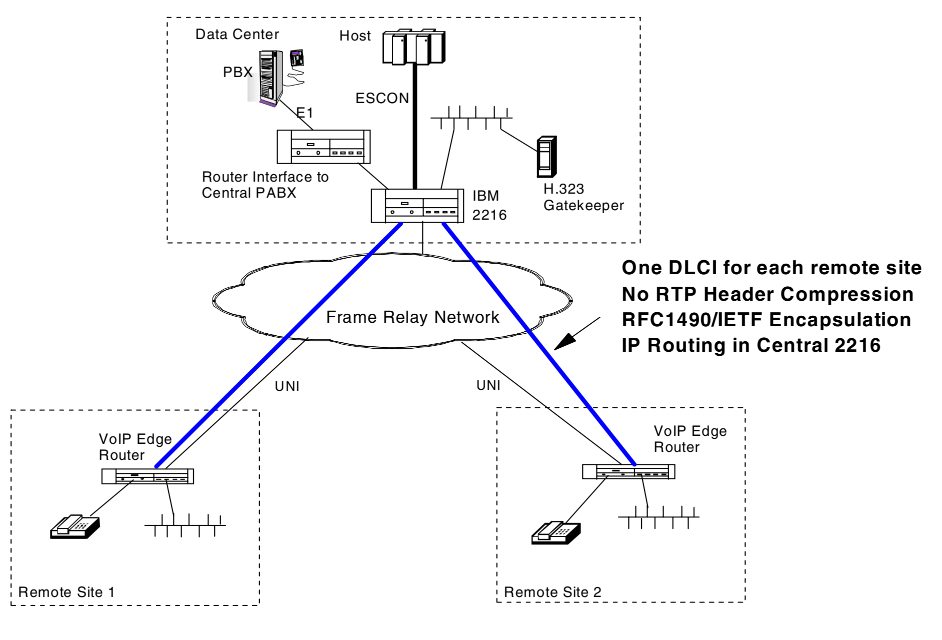

Data Center
Host
PBX
ESCON
E1
Router Interface to Central PABX
IBM 2216
H.323 Gatekeeper
Frame Relay Network
One DLCI for each remote site
No RTP Header Compression
RFC1490/IETF Encapsulation
IP Routing in Central 2216
UNI
UNI
VoIP Edge Router
VoIP Edge Router
Remote Site 1
Remote Site 2

Figure 33. Voice over IP in a frame relay network - single DLCI with no RTP compression

Each remote site Cisco router is configured to support IETF frame encapsulation and to transmit voice and data traffic in uncompressed form using a single DLCI. The IBM router is simply performing IP routing on all packets and has no direct telephony interfaces. The configurations presented here are based on the complete IP routing network reviewed in Chapter 2, “Dynamic routing protocols” on page 13.

Only the configuration steps relevant to supporting voice traffic are reviewed here in detail.

The IBM router must be configured to:

- Fragment large data packets using FRF.12 fragmentation.
- Provide VoIP RTP priority using BRS (DiffServ is incompatible with FRF.12).
- Perform frame relay traffic shaping through the CIR Monitor with appropriate CIR and B_c settings.

The relevant Cisco configuration statements required in one of the remote site Cisco routers to implement this network are shown in Figure 34 on page 73.

```
!
voice-port 1/0/0
 timeouts call-disconnect 0
 description connected to Phone (321321)
!
voice-port 1/0/1
 timeouts call-disconnect 0
 description connected to Phone_3 (321322)
!
dial-peer voice 1 pots
 destination-pattern 321321
 port 1/0/0
!
dial-peer voice 3 voip 1
 destination-pattern 32131.
 ip precedence 5 2
 session target ipv4:10.1.11.3
!
dial-peer voice 2 pots
 destination-pattern 321322
 port 1/0/1
!
num-exp 1.. 3213..
!
interface Serial0/0
 mtu 2044 3
 ip address 10.1.4.10 255.255.255.0 4
 no ip directed-broadcast
 encapsulation frame-relay IETF 5
 ip ospf network non-broadcast
 ip ospf hello-interval 10
 no ip mroute-cache
 cdp enable
 frame-relay traffic-shaping 6
 frame-relay interface-dlci 220 7
  class voice 8
 frame-relay lmi-type ansi
 ip rtp priority 16384 100 24 9
!
map-class frame-relay voice 10
 no frame-relay adaptive-shaping
 frame-relay cir 32000 11
 frame-relay bc 1920 12
 frame-relay be 0
 frame-relay fair-queue 13
 frame-relay fragment 80 14
```

Figure 34. Cisco Sydney router configuration to support single PVC with no RTP compression

The key configuration parameters are:

- Connections to remote voice peers will use VoIP. 1

- IP precedence is assigned to the TOS bits in all VoIP frames so that any routers along the path can use this setting to prioritize voice traffic. 2
- MTU size is set to match IBM defaults. Fragmentation will be used to reduce the size of large data frames. 3
- The appropriate IP address is assigned to this interface for all data and VoIP frames to use. 4
- RFC1490/IETF encapsulation is configured to allow compatibility with the IBM router. 5
- Frame relay traffic shaping is applied to the interface to make sure that all traffic conforms to CIR settings etc., so that voice frames are not dropped by the frame relay switches. 6
- A frame relay PVC is mapped to this interface. This PVC will carry voice and data IP traffic. 7
- The frame relay map class called “voice” is applied to this particular PVC. 8
- The `ip rtp priority` command is applied to ensure that RTP voice traffic has absolute priority. 9
- The frame relay map class called “voice” is defined. 10
- The CIR and committed burst size are set. 11 and 12
- The WFQ scheme is applied to the map class. 13
- FRF.12 fragmentation is applied to the map class to allow large data frames to be fragmented. 14

3.1.4.5 VoIP example using Cisco RTP header compression

To use voice over IP in a low-speed frame relay WAN environment, it is necessary to utilize Cisco’s proprietary RTP header compression technique to achieve bandwidth efficiency. In this situation interoperability is still possible, for example between Cisco branch routers and IBM central routers, but only by using the IBM router as a frame relay switch. The IBM router’s frame relay frame handler (FRFH) support is used to switch all frames between voice-capable Cisco routers. In a large network environment all traffic from voice-capable remote site Cisco routers will be switched to a central Cisco router for decoding as shown in Figure 35 on page 75. The central site Cisco router will then connect to a central site PABX or it will switch voice calls to another remote site as required.This implementation uses a single DLCI for each remote site and all traffic from these sites is encapsulated in Cisco’s proprietary frame relay format that supports RTP header compression. The central Cisco router may be configured to route all data frames back to the IBM 2216 for transmission to the mainframe channel or to another non-voice site. This approach is reasonable where the majority of remote sites are not voice-capable and a single frame relay access service is required at the central site. If the majority of remote sites require VoIP support then it would be more efficient to connect the central Cisco router directly to the frame relay service and to the LAN and to utilize the IBM 2216 simply as a host gateway service.

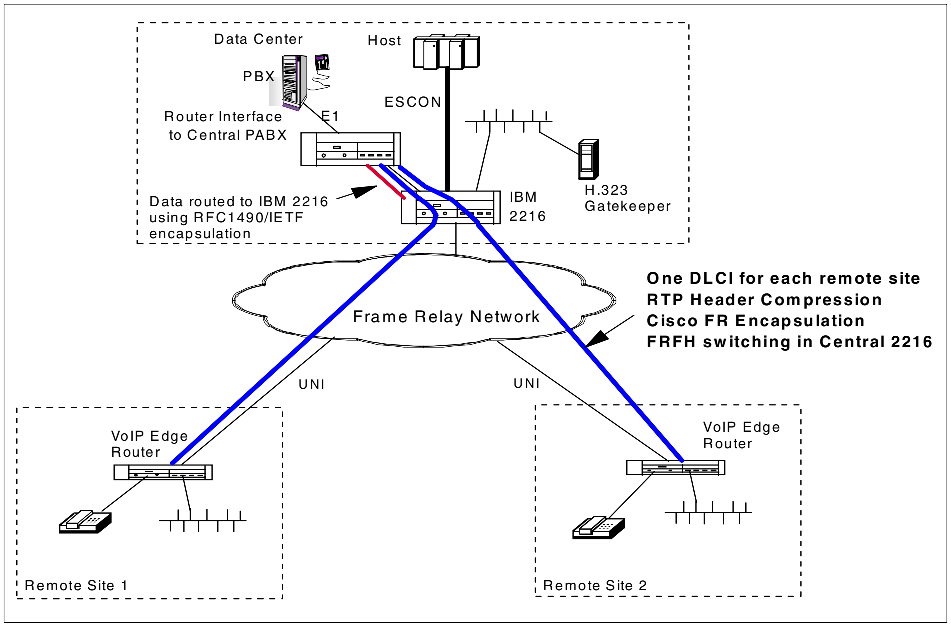


Figure 35. FRFH function in IBM 2216 switching remote site traffic to Cisco central router

Implementation sample

The Cisco and IBM router configurations required to implement this approach are now reviewed. A simplified configuration was implemented based on the network shown in Figure 36 on page 76. This configuration uses a single DLCI per remote site and all remote site traffic uses Cisco proprietary frame relay encapsulation that supports RTP header compression. All traffic must therefore be switched via the FRFH function of the IBM 2216. In this example we switched all traffic directly between the two Cisco 2621 remote site routers directly, rather than to a central Cisco router.

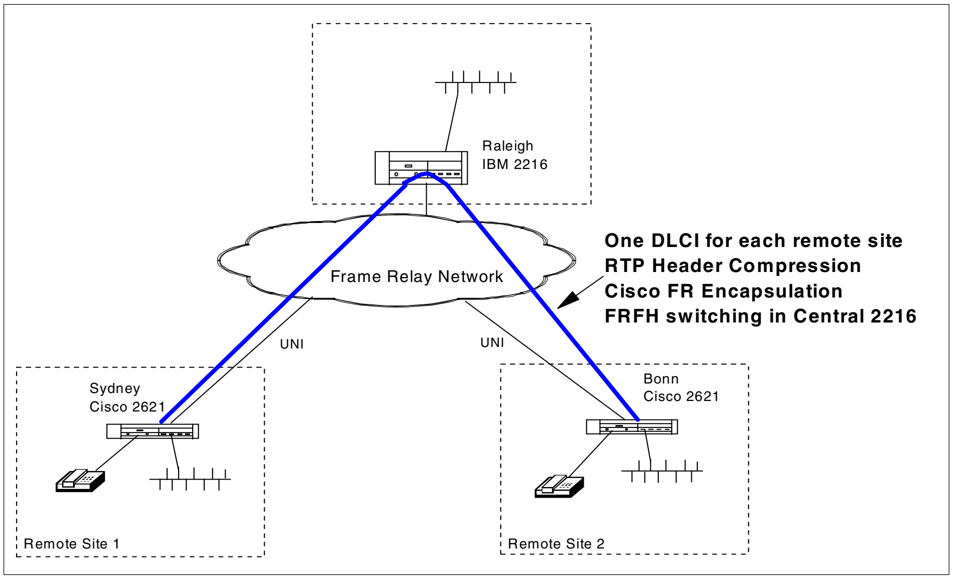


Figure 36. Voice over IP connection between Cisco 2621 router via IBM 2216

The Cisco 2621 routers are configured to use VoIP with RTP compression providing an efficient voice link through the frame relay network. Telephone handsets are attached to FXS ports on the Cisco 2600 voice interface card in each 2621 router. The IBM 2216 performs frame relay switching using its FRFH function.

The Sydney Cisco 2621 configuration statements relating to voice support are shown in Figure 37 on page 77.

```
!
voice-port 1/0/0
 timeouts call-disconnect 0
 description connected to Phone (321321)
!
voice-port 1/0/1
 timeouts call-disconnect 0
 description connected to Phone_3 (321322)
!
!
dial-peer voice 1 pots
 destination-pattern 321321
 port 1/0/0
!
dial-peer voice 3 voip
 destination-pattern 32131.
 ip precedence 5
 session target ipv4:10.1.11.3
!
dial-peer voice 2 pots
 destination-pattern 321322
 port 1/0/1
!
num-exp 1.. 3213..

interface Serial0/0
 mtu 2044
 ip address 10.1.4.10 255.255.255.0
 no ip directed-broadcast
 encapsulation frame-relay 1
 ip ospf network non-broadcast
 ip ospf hello-interval 10
 no ip mroute-cache
 frame-relay traffic-shaping
 frame-relay interface-dlci 120
  class voice
 frame-relay ip rtp header-compression 2
 ip rtp priority 16384 100 24
!
map-class frame-relay voice
 no frame-relay adaptive-shaping
 frame-relay cir 32000
 frame-relay bc 1920
 frame-relay be 0
 frame-relay fair-queue
 frame-relay fragment 80
```

Figure 37. Sydney Cisco 2621 configuration for VoIP using RTP header compression

This configuration is very similar to the Cisco configuration shown in Figure 34 on page 73.

The differences are:

- Cisco frame relay encapsulation is applied to the interface. This is mandatory to support the proprietary RTP header compression over frame relay. 1
- RTP header compression is enabled. This means that any IBM routers in the data path must be configured to switch all frame relay traffic on to another Cisco router that will be able to interpret the compressed traffic.

3.2 Voice over frame relay

Voice transport over frame relay networks is now a mature technology. Many frame relay access device (FRAD) and router vendors have implemented voice interface adapters and efficient voice compression algorithms to provide reliable and high-quality voice transport over frame relay WANs.

Some of the relevant standards in this area include:

- FRF.11 - The frame format that supports the transport of a large number of voice channels in a single DLCI and transports the signalling required to implement a dial plan across an organization.
- FRF.12 - The fragmentation technique that allows large data frames to be fragmented and interleaved with time-critical small voice frames. FRF.12 fragmentation of large data frames allows a router to maintain acceptable interframe gaps and latency of voice frames and improve the resulting voice quality.
- Vocoder compression standards including:
  - G.726 - ADPCM at 32 kbps
  - G.728 - LDCELP at 16 kbps
  - G.729 - CSCELP at 8 kbps

Most standards-based implementations over low-speed frame relay links use G.729 to achieve high-quality voice communications using approximately 12 kbps bandwidth per voice call. The additional bandwidth (8 kbps --> 12 kbps) is due to the overhead of the frame relay headers.

3.2.1 Voice over frame relay network components

This section reviews the connectivity options and components required to allow voice traffic to be transported natively over a frame relay network. A basic configuration is shown in Figure 19.

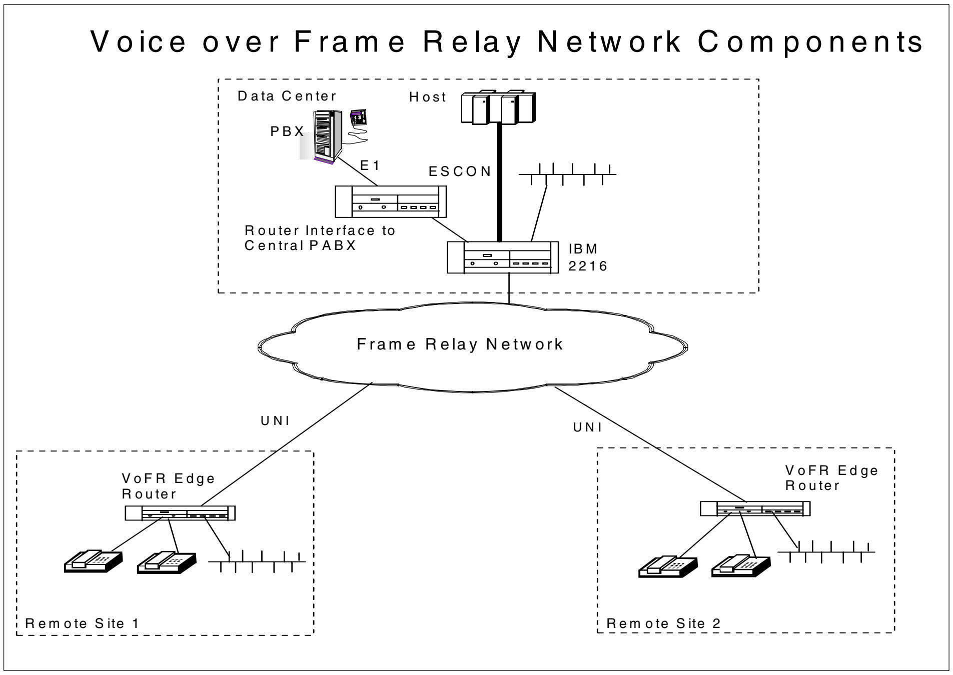


Figure 38. Voice over frame relay network components

The principle components of this configuration are:

- Edge routers, which provide direct voice equipment interfaces. These routers digitize and compress the voice traffic and transmit it through the frame relay WAN using FRF.11 frame format. Cisco routers support VoFR traffic in either FRF.11 frame format or in a proprietary Cisco format.
- Core network routers. Any intermediate router that is not FRF.11 compatible must be configured to simply switch FRF.11 frames between frame relay PVCs.
- Central voice routers, which provide high-speed digital interfaces via E1 or T1 to digital PABXs and are able to interpret FRF.11 frames and switch calls between frame relay PVCs to the correct destination. These routers are also responsible for Tandem PABX bypass, which means that voice

calls between two remote sites do not have to enter the central PABX for call switching. The central router performs digital switching based on the FRF.11 call signalling information.

3.2.2 Voice over frame relay interoperability - IBM and Cisco routers

The VoFR implementation delivered by IBM in June 1999 allows voice frames and data frames to share a single frame relay PVC by encapsulating voice frames in FRF.11 format and data frames in RFC1490/IETF format in the same frame relay PVC. FRF.11 frames are encoded and decoded directly by the voice interface modules of the IBM 2212 router. The IBM router common code actually only constructs and interprets frames in RFC1490/IETF format and must be configured to switch FRF.11 frames either to another frame relay PVC or to pass them to a voice interface module inside the IBM 2212.

Cisco routers, on the other hand, require that each frame relay PVC be configured with a single frame relay encapsulation type and will therefore not allow FRF.11 and RFC1490/IETF frames to share a single PVC. This means that there are two options for configuring a mixed IBM/Cisco VoFR network, both of which are described in the following sections. The first option requires separate frame relay PVCs to each site for voice and data and is reviewed in 3.2.3, “Voice over frame relay using multiple PVCs - IBM and Cisco” on page 86. The second option requires only a single frame relay PVC per remote site and is reviewed in 3.2.4, “Voice over frame relay using a single PVC - IBM and Cisco” on page 92.

In each case we assume that the network includes an existing IBM 2216 central office router with a frame relay WAN providing connectivity to a number of remote sites in which Cisco routers may be installed to provide voice support. The issues to consider when adding voice support to the network are reviewed. First we document the configuration parameters that must be specified on IBM routers to provide QoS in this environment and the corresponding Cisco router configuration parameters to provide compatibility.

3.2.2.1 QoS configuration for VoFR - IBM and Cisco routers

A number of parameters need to be configured in a mixed VoFR network of IBM and Cisco routers to ensure high-quality voice communications. In all cases we assume that Cisco routers will provide the telephony equipment interfaces and introduce the voice traffic to the network.

In this situation both IBM and Cisco routers need to be configured to:

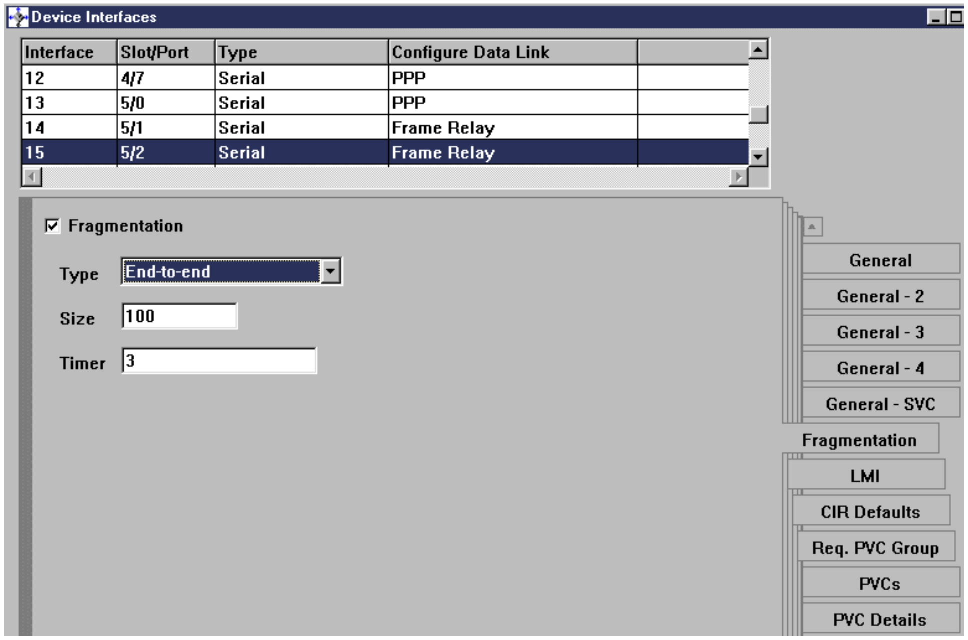
- Fragment large data frames using FRF.12 or FRF.11(Cisco) fragmentation

- Perform frame relay traffic shaping by enabling CIR monitor and setting the correct CIR and B_c parameters.
- Prioritize voice traffic over data traffic within FR PVCs

Configuring fragmentation and frame relay traffic shaping

IBM routers only support FRF.12 fragmentation for frame relay links.

FRF.12 fragmentation must first be enabled on IBM routers for the entire interface, using the panel shown in Figure 39.



Interface	Slot/Port	Type	Configure Data Link	
12	4/7	Serial	PPP	
13	5/0	Serial	PPP	
14	5/1	Serial	Frame Relay	
15	5/2	Serial	Frame Relay	

☑ Fragmentation

Type End-to-end

Size 100

Timer 3

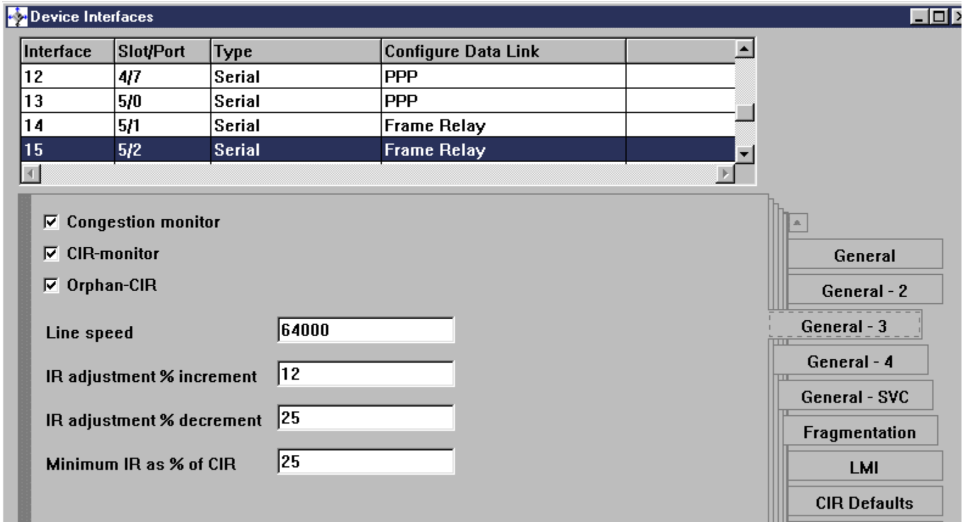
Figure 39. Enable FRF.12 fragmentation on IBM routers

Fragmentation must first be enabled here at the physical link level before it can be specified on any individual PVC. The default fragment type of UNI/NNI should normally be changed to End-to-End unless the frame relay service provider supports UNI/NNI fragmentation (which is extremely unlikely; see “Fragmentation and interleaving on frame relay links” on page 47). Again the fragment size must be specified to suit the link speed, generally 100 bytes or less on a 64 kbps link. The “timer” parameter specifies how many seconds to wait for the next fragment in a sequence before discarding fragments already received.

Note - FRF.12 fragmentation

When using the IBM graphical configurator for IBM routers, if end-to-end fragmentation has been specified on IBM routers, and fragmentation parameters have been set at the link level, it is also necessary to enable fragmentation explicitly and to set the fragment size for each PVC on the link. If you do not specify these parameters for each PVC then fragmentation will not occur on those PVCs and communication with Cisco routers using FRF.12 fragmentation will fail.

To enable frame relay traffic shaping on the IBM routers you must enable the IBM router's CIR monitor and then set appropriate values for the CIR and the committed burst size (B_c) fields, as shown in Figure 40 and Figure 41.



Interface	Slot/Port	Type	Configure Data Link	
12	4/7	Serial	PPP	
13	5/0	Serial	PPP	
14	5/1	Serial	Frame Relay	
15	5/2	Serial	Frame Relay	

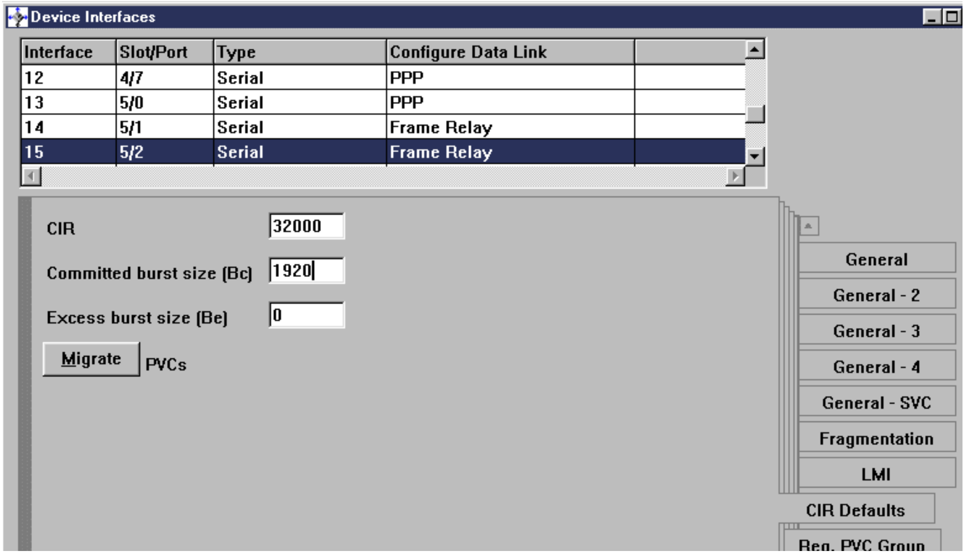
☑ Congestion monitor

☑ CIR-monitor

☑ Orphan-CIR

Line speed	64000
IR adjustment % increment	12
IR adjustment % decrement	25
Minimum IR as % of CIR	25

Figure 40. Enable CIR monitor on IBM routers



Interface	Slot/Port	Type	Configure Data Link
12	4/7	Serial	PPP
13	5/0	Serial	PPP
14	5/1	Serial	Frame Relay
15	5/2	Serial	Frame Relay

CIR 32000

Committed burst size (Bc) 1920

Excess burst size (Be) 0

Figure 41. Set CIR and B_c values to govern traffic shaping

The CIR monitor forces the router to honor the settings that should match those of the network itself and reduces the chance of any frames being dropped by the network. As shown in Figure 41, these parameters may be specified as defaults for all PVCs on an IBM router interface, although it is possible to override the defaults and configure specific CIR and B_c values for each PVC.

For a more complete review of appropriate setting for CIR and B_c refer to 3.1.4.3, “General considerations for setting frame relay parameters” on page 70.

Cisco routers have three possible fragmentation types:

- FRF.12 fragmentation for data-only PVCs. FRF.12 fragmentation is also recommended by Cisco when carrying voice over IP traffic in a frame relay network.
- FRF.11 fragmentation is used if a particular PVC is carrying both voice and data traffic in FRF.11 frames.
- Proprietary Cisco fragmentation, which was originally implemented in the Cisco 3810 and would *not* normally be used in a mixed IBM and Cisco VoFR network.

Even though IBM routers do not directly support FRF.11 fragmentation, this does not present a compatibility problem, since all traffic in these PVCs will be voice traffic in FRF.11 encapsulated frames, which must be switched through the IBM router using the FRFH function. If separate PVCs are used for voice and data in the Cisco router, then the data-only PVCs will be configured to use RFC1490/IETF encapsulation and FRF.12 fragmentation, which is understood by the IBM router.

Fragmentation and frame relay traffic shaping are configured on Cisco routers as part of a frame relay map class, which sets the frame relay characteristics to be applied to PVCs.

A sample frame relay map-class is shown in Figure 42.

```
interface Serial0/0
 Frame-relay traffic-shaping 1
 frame-relay lmi-type ansi
 no ip directed-broadcast
 frame-relay interface-dlci 172 2
  class VoicePVC 3
  vofr data 4 call-control 5 4
!
map-class frame-relay VoicePVC 5
 frame-relay cir 32000
 frame-relay bc 1920
 frame-relay mincir 16000
 frame-relay fair-queue 6
 frame-relay voice bandwidth 12000 queue 7
 frame-relay fragment 120 8
```

Figure 42. Map class definition for Cisco frame relay Interface

- Frame relay traffic shaping is applied to the base interface .1
- A frame relay interface is defined and mapped to DLCI number 172. 2
- A map class is assigned to this PVC - we called the map class "VoicePVC". 3
- The `vofr` command enables this PVC for voice traffic using FRF.11 encapsulation. Data will use subchannel 4 and call control(signalling) will use subchannel 5. 4
- A map class is defined - we called this map class "VoicePVC" as it specifies suitable frame relay parameters to carry voice. 5
- The Weighted Fair Queueing (WFQ) prioritization and queuing scheme is applied. 6

- Voice bandwidth is allocated and the special “queue” parameter is set (only available in IOS 12.0(5)T and above) to give voice absolute priority over data in the PVC. **7**
- Fragmentation size is set (FRF.11 in this case, since this is a PVC that carries voice). If the PVC was not configured for voice then FRF.12 fragmentation would be used automatically by the Cisco router. **8**

Note

- Unless `frame-relay traffic-shaping` is applied to the base interface, VoFR will not work on the Cisco router.
- The `vofr` command must have both data and call-control subchannels specified if you want to allow for switched calls using FRF.11.
- If the `fragmentation` parameter is enabled, Cisco routers automatically use FRF.12 for data-only PVCs and FRF.11 fragmentation for PVCs carrying both voice and data.

Configuring voice traffic prioritization

IBM routers offer two prioritization and bandwidth reservation schemes for traffic,:Bandwidth Reservation System (BRS) and DiffServ. VoFR traffic is not IP traffic; therefore DiffServ is not applicable to this environment. Furthermore, it will not normally be necessary to configure BRS on the IBM routers in this environment because BRS only prioritizes and schedules traffic *within* a frame relay PVC or DLCI. BRS can be configured to assign different bandwidth percentages to separate PVCs, but as long as the sum of PVC CIRs on an interface does not exceed the access speed of the interface, then the bandwidth percentages assigned to the BRS circuit classes are not used. The FR traffic shaping function (that is, the CIR monitor) will override the BRS circuit class bandwidth allocations in this situation. Therefore BRS would normally have no effect on prioritizing or scheduling traffic *between* PVCs.

PVCs carrying FRF.11 traffic, including voice traffic, will always be switched by IBM routers using the frame relay frame handler (FRFH) function of IBM routers, therefore BRS will never have any role in prioritizing traffic within these PVCs either.

If separate frame relay PVCs are configured between IBM and Cisco routers to carry data using RFC1490/IETF encapsulation, then BRS could be configured to prioritize different types of data traffic within the RFC1490/IETF data PVCs. For example, SNA data could be prioritized over IP FTP traffic, but this has no impact on the forwarding of any voice traffic using FRF.11 frames through the router, since these will use separate PVCs.

CIsco routers offer a number of queueing and traffic prioritization schemes, which are summarized in 3.1.2.3, “Prioritization of VoIP traffic in IBM and Cisco routers” on page 53. Note, however, that Weighted Fair Queuing (WFQ) is the only queuing mechanism supported on a PVC when fragmentation is configured on the PVC, so WFQ will normally be the only option in a low-speed frame relay environment.

WFQ establishes an implicit high-priority queue for use by time-sensitive traffic such as voice frames. The new (IOS 12.0(5)T) command:

```
frame-relay voice bandwidth <bps> queue <depth>
```

allocates VoFR voice frames to this absolute priority queue in a PVC that is carrying both voice and data.

On high speed frame relay links, where fragmentation is not required, class-based weighted fair queuing (CBWFQ) may be used to further classify traffic.

3.2.3 Voice over frame relay using multiple PVCs - IBM and Cisco

This configuration supports full data and routing interoperability between IBM and Cisco routers while supporting the addition of voice over frame relay traffic to the network. This approach requires that multiple frame relay PVCs be configured to each remote site to allow data to be carried via RFC1490/IETF frames in one PVC and voice via FRF.11 frames in another PVC, as shown in Figure 43 on page 87. This approach allows data traffic from all existing remote sites as well as the new voice-capable sites to be routed directly to the existing IBM 2216 via a single central site frame relay access interface. The voice traffic is then essentially overlaid on this network using new PVCs and is switched through the central IBM 2216 via the IBM router frame relay frame handler (FRFH) support, allowing direct communication between the central and remote site voice-capable Cisco routers.

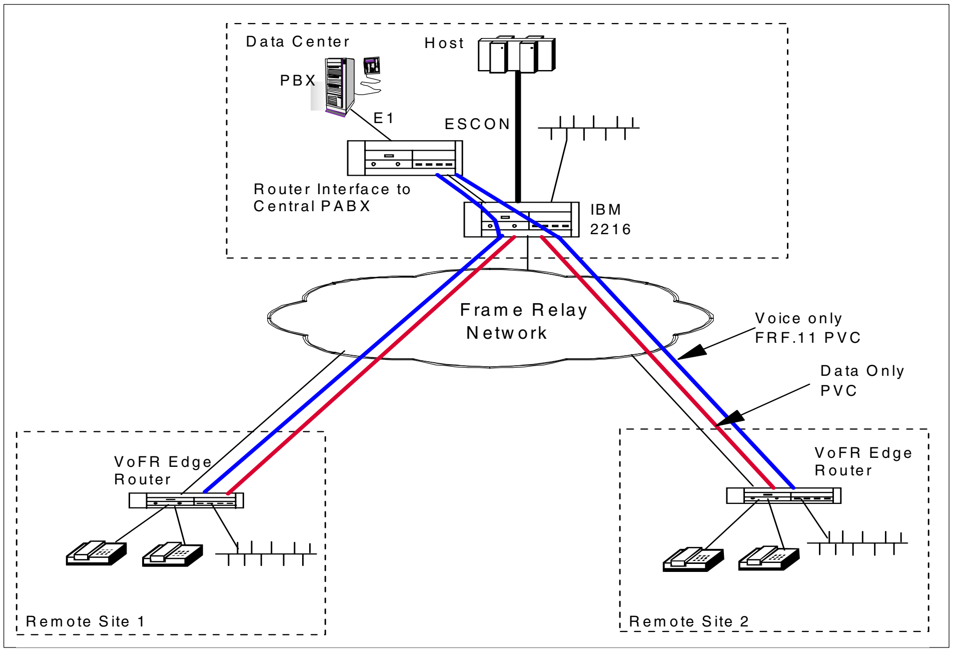


Figure 43. Voice over frame relay using multiple PVCs - IBM and Cisco routers

Implementation sample

The Cisco and IBM router configurations required to implement this approach are now reviewed. A simplified configuration was implemented based on the network shown in Figure 44.

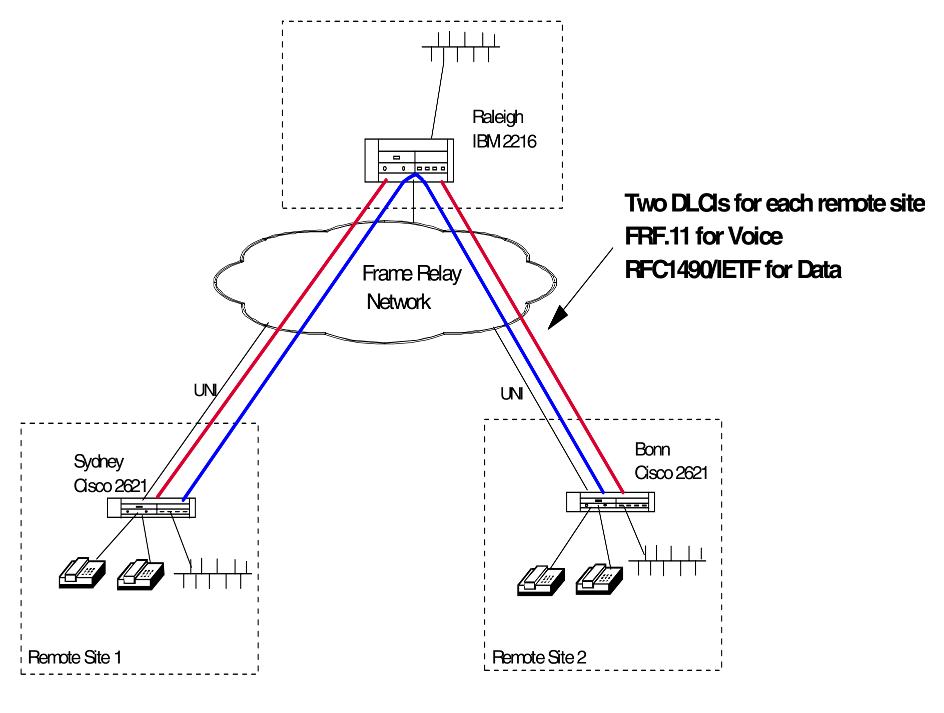


Figure 44. Voice over frame relay using multiple DLCIs between IBM and Cisco routers

This configuration switches the voice PVC directly between the two remote site Cisco routers. Relevant sections of the Sydney Cisco 2621 router configuration required to implement the multiple DLCIs and encapsulation formats are shown in Figure 45 on page 89. This example shows how to configure a Cisco router to use multiple frame relay PVCs with RFC1490/IETF encapsulation on some PVCs for data transport and FRF.11 encapsulation on others for voice transport.

```
voice-port 1/0/0
 timeouts call-disconnect 0
 description connected to Phone (321321)
!
voice-port 1/0/1
 timeouts call-disconnect 0
 description connected to Phone_3 (321322)
!
dial-peer voice 1 pots
 destination-pattern 321321
 port 1/0/0
!
dial-peer voice 2 pots
 destination-pattern 321322
 port 1/0/1
!
dial-peer voice 3 vofr 1
 destination-pattern 32131.
 session target Serial0/0 120
!
num-exp 1.. 3213..
!
interface Serial0/0
mtu 2044 2
no ip address
no ip directed-broadcast
encapsulation frame-relay IETF 3
frame-relay traffic-shaping 4
frame-relay lmi-type ansi
!
interface Serial0/0.3 multipoint 5
ip address 10.1.4.10 255.255.255.0
no ip directed-broadcast
ip ospf network non-broadcast
ip ospf hello-interval 10
frame-relay interface-dlci 104 6
class nonvoice 7
 frame-relay interface-dlci 220
class nonvoice
!
interface Serial0/0.120 point-to-point 8
no ip directed-broadcast
frame-relay interface-dlci 120 9
class voicepvc 10
vofr data 4 call-control 5 11

! NOTE - Required MAP Class definitions are shown in Figure 46 on page 91
```

Figure 45. Sydney Cisco router configuration

The key points in this configuration are:

- Indicates in dial plan that target is VoFR rather than VoIP. 1
- Leaves MTU size large and use FRF.12 to fragment large data frames. 2
- Base serial interface has RFC1490/IETF encapsulation set for IBM compatibility. 3
- Enables frame relay traffic shaping on base interface - mandatory for VoFR. 4
- First subinterface configured takes base IETF encapsulation for data. 5
- Assigns DLCIs 104 and 220 to this frame relay subinterface for data transfer to 2216 or other frame relay connected routers in the network using RFC1490/IETF encapsulation. 6
- Applies a map class that we called *nonvoice*, to assign traffic shaping parameters to this DLCI. 7
- Frame relay subinterface for voice traffic transport. 8
- Assigns the correct DLCI (120) to this subinterface. This PVC will be switched through the IBM 2216 router using the FRFH function. 9
- Applies a map class to this interface that assigns appropriate traffic shaping characteristics for voice traffic. 10
- The `vofr` command specified FRF.11 format with data in subchannel 4 and call control information in subchannel 5. 11

The frame relay map classes that are also required as part of this configuration are shown in Figure 46 on page 91.

```
map-class frame-relay voicepvc 1
 no frame-relay adaptive-shaping
 frame-relay cir 16000 2
 frame-relay bc 1920 3
 frame-relay be 0
 frame-relay mincir 16000
 frame-relay fair-queue 4
 frame-relay voice bandwidth 16000 5
 frame-relay fragment 80 6
!
map-class frame-relay nonvoice 7
 no frame-relay adaptive-shaping
 frame-relay cir 16000
 frame-relay bc 1920
 frame-relay be 0
 frame-relay fair-queue 8
 frame-relay fragment 80 9
```

Figure 46. Map class definition for voice over frame relay implementation

The key points in the map class definition are:

- Defines a map class with traffic shaping parameters suitable for FRF.11 voice traffic. 1
- Sets CIR matching the true frame relay PVC CIR. 2
- Sets the committed burst size to allow a voice frame and at least some data fragments to be sent in the burst interval. 3
- Applies Weighted Fair Queuing to the class. This is mandatory to allow prioritization of voice traffic. 4
- Allocates priority voice bandwidth within WFQ. 5
- Sets the fragmentation size (80 bytes is good for 64 kbps links). Since this PVC is carrying voice and data via FRF.11, then the fragmentation type is automatically set to FRF.11 fragmentation by the router. 6
- Defines a new map class to apply to PVCs that are not carrying voice traffic. 7
- WFQ must also be applied to this definition, since it is the only queuing strategy supported if fragmentation is required. 8
- Sets the fragment size. Since this map class will be applied to non-voice PVCs, the fragmentation type will be FRF.12. This is compatible with IBM routers. 9

3.2.4 Voice over frame relay using a single PVC - IBM and Cisco

This configuration allows the remote site Cisco router to carry all traffic via FRF.11 frames in a single DLCI. This approach reduces the number of frame relay PVCs required, since the remote site Cisco routers will send all voice and data in a single PVC using FRF.11 encapsulation. The central IBM router is not able to interpret FRF.11 frames directly and must be configured to switch all these frames to a centrally located Cisco router, which interprets the FRF.11 information and performs the necessary voice switching. This approach may be a little more cost-effective than the previous multiple PVC approach for some organizations, although the total CIR requirement for the site will be about the same in both cases. This configuration is shown in Figure 47.

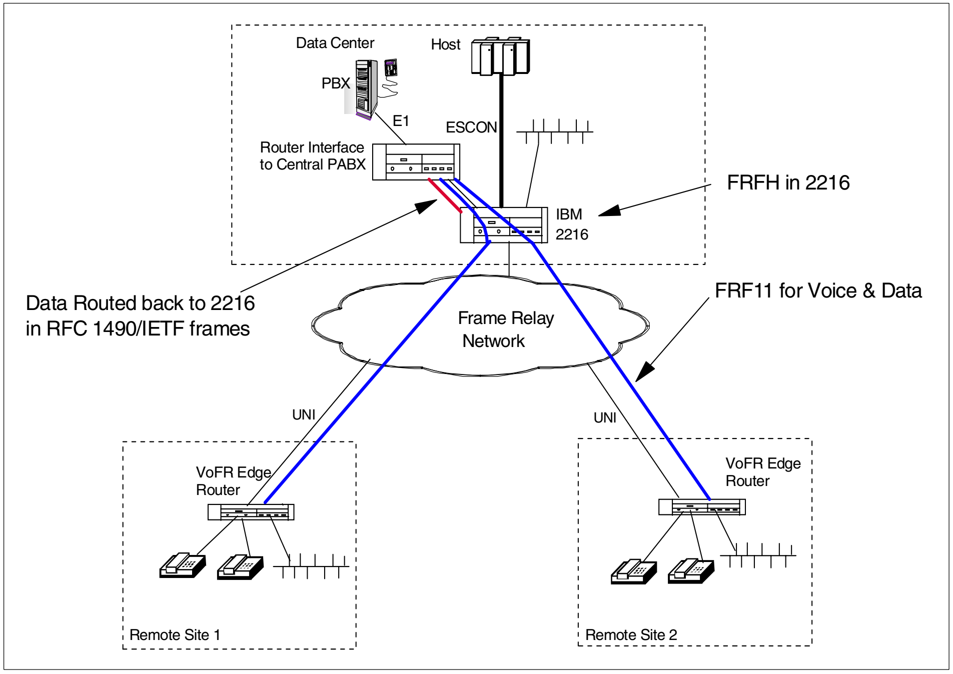


Figure 47. Single frame relay PVC configuration utilizing FRFH function of IBM 2216

This approach reduces the number of PVCs required, but means that existing data-only remote site branches will use the IBM 2216 as their central router, whereas Cisco voice-enabled branches will use the central Cisco router as their central data router. Frames must then be routed back to the IBM 2216 in RFC 1490/IETF format for routing to the S/390 or other sites.

Implementation sample

The Cisco and IBM router configurations required to implement this approach are now reviewed. A simplified configuration was implemented based on the scenario shown in Figure 48. This example shows how to configure a Cisco router to use a single frame relay PVC for FRF.11 formatted voice and data traffic.

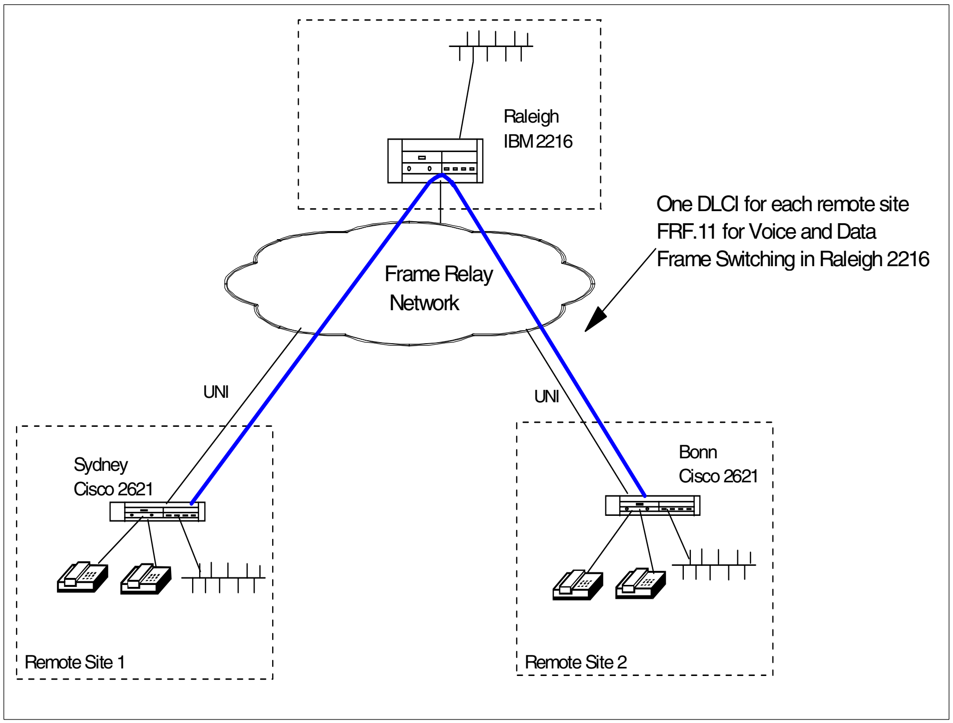


Figure 48. Voice over frame relay using one DLCI between IBM and Cisco routers

The Sydney Cisco router configuration showing those steps required to configure transport of voice and data traffic via FRF.11 encapsulation is shown in Figure 49. This is not the complete Cisco router configuration, since we are specifically addressing parameters relating to FRF.11 voice transport here. Complete Cisco router configurations supporting the underlying IP data routing environment are reviewed in Chapter 2, “Dynamic routing protocols” on page 13.

```
!
voice-port 1/0/0
 timeouts call-disconnect 0
 description connected to Phone (321321)
!
voice-port 1/0/1
 timeouts call-disconnect 0
 description connected to Phone_3 (321322)
!
dial-peer voice 1 pots
 destination-pattern 321321
 port 1/0/0
!
dial-peer voice 2 pots
 destination-pattern 321322
 port 1/0/1
!
dial-peer voice 3 vofr 1
 destination-pattern 32131.
 session target Serial0/0 120
!
num-exp 1.. 3213..

interface Serial0/0
 mtu 2048 12
 bandwidth 64
 ip address 10.1.4.10 255.255.255.0 3
 no ip directed-broadcast
 encapsulation frame-relay
 ip ospf network point-to-point
 no ip mroute-cache
 frame-relay traffic-shaping 4
 frame-relay interface-dlci 120 5
  class voicepvc 6
  vofr data 4 call-control 5 7
 frame-relay lmi-type ansi
!
map-class frame-relay voicepvc 8
 no frame-relay adaptive-shaping
 frame-relay cir 16000 9
 frame-relay bc out 16000
 frame-relay be out 16000
 frame-relay mincir 16000
 frame-relay fair-queue 10
 frame-relay voice bandwidth 16000 11
 frame-relay fragment 80 12
```

Figure 49. Sydney Cisco router configuration - single FRF.11 DLCI

The key points in this configuration are:

- Indicates in dial plan that target is VoFR rather than VoIP. 1
- Leaves MTU size large and uses fragmentation to fragment large data frames. 2
- The IP address is assigned to this interface. IP data sent to this interface will be encapsulated in FRF.11 frames. 3
- Enables frame relay traffic shaping on base interface - mandatory for VoFR. 4
- First subinterface configured takes base IETF encapsulation for data.5
- Assigns subinterface to DLCI for data and voice transfer. 6
- Applies a map class that we called *voicepvc*, to assign traffic shaping parameters to this DLCI. 7
- Assigns data traffic to subchannel 4 and call control to subchannel 5. 8
- Sets CIR to match the true CIR for this PVC. 9
- Applies WFQ as the queueing scheme. This is the only one supported in a fragmentation environment. 10
- Assigns voice to a priority queue and allocate the required bandwidth. 11
- Assigns an appropriate fragment size. Since this map class will be assigned to PVCs carrying voice and data, the fragmentation scheme will be FRF.11 fragmentation. 12

Chapter 4. APPN interoperability and migration

This chapter reviews the APPN functions available with Cisco routers and examines interoperability with IBM router APPN functions. Many organizations operate large SNA networks based on the APPN routing functions available in Communications Server for OS/390 (CS for OS/390), the 3745, 3746-9x0 Network Node and the IBM 221x range of routing products. These products all offer APPN functions including:

- Complete Network Node implementation, including topology database and locate functions
- Enterprise Extender - APPN/HPR over UDP/IP
- Dependent LU server/dependent LU requester[1] (DLUS/DLUR) to support legacy LU2 (3270 screen and printer) and other dependent sessions over an APPN network
- Branch Extender - a hybrid Network Node/End Node limiting the amount of APPN topology update broadcasts in very large networks

Cisco has offered APPN functions on its router platform since 1995. A new APPN/SNA subsystem has become available in IOS 12.1 which replaces the original Cisco APPN implementation. This new subsystem is called SNA Switching Services (SNASw) and is configured using a new command set. This subsystem is oriented more towards a network that implements an IP infrastructure while still supporting SNA applications. It is designed to be simpler to configure and operate and places less APPN function in the network than the original APPN subsystem. Cisco IOS releases from 12.1 onwards no longer support the APPN command set; all APPN functions are now configured with the `SNASw` command set.

Please refer to other literature if you are not already familiar with terms and concepts such as APPN (Advanced Peer-to-Peer Networking), NN (Network Node), EN (End Node), LEN (Low-Entry Networking), DLUR (Dependent LU Requester) and HPR (High Performance Routing). Some excellent references are:

- *Inside APPN and HPR - The Essential Guide to New SNA*, SG24-3669
- *Subarea to APPN Migration: HPR and DLUR Implementation*, SG24-5204

[1] DLUS is only implemented by VTAM running on OS/390, VM or VSE operating systems

4.1 Cisco SNA Switching Services

SNA Switching Services (SNASw) is an implementation of a router as a Branch Network Node (BrNN) according to the APPN Branch Extender architecture. A Cisco router running the SNASw subsystem does not implement traditional Network Node function; it *appears* to be an APPN Network Node to “downstream” devices while at the same time appearing to be an APPN End Node to “upstream” devices. This function is represented in Figure 50 on page 98. While a Branch Network Node cannot be placed in identical configurations as a full Network Node (some network redesign may be required), the SNASw subsystem implements a wide range of APPN connectivity options and, as we show, proves to be compatible with IBM’s APPN implementation. Any network redesign may well be worth the effort because it will increase the stability and reliability of the network; Cisco’s SNASw is also very simple to configure.

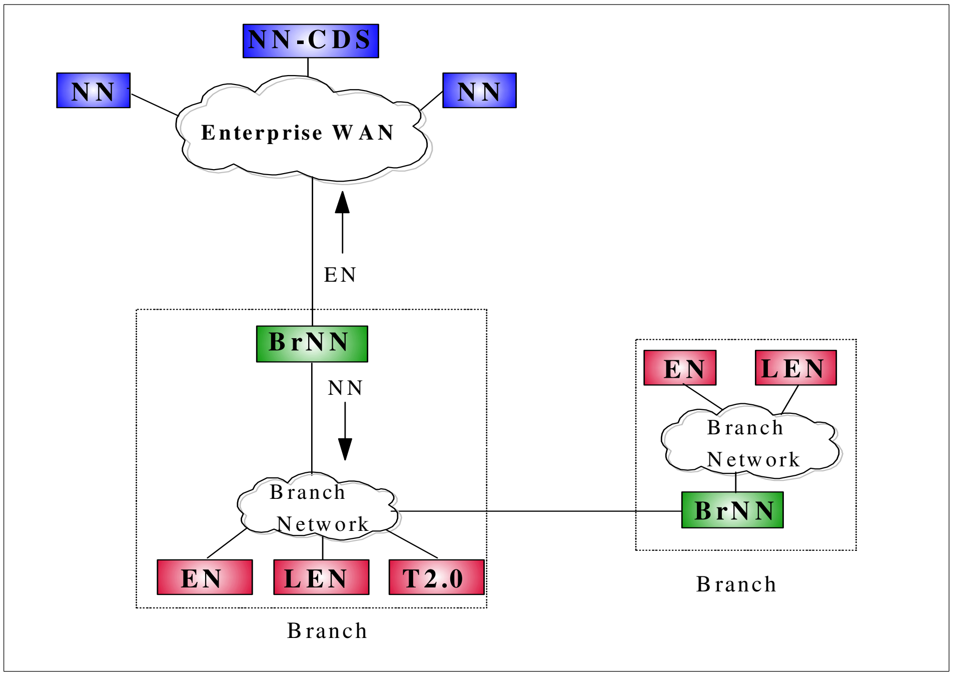


Figure 50. Branch Extender implemented by Branch Network Nodes

Upstream and Downstream

In this chapter, the term “downstream” refers to the branch network in which the BrNN has an appearance of an APPN Network Node whereas “upstream” refers to the network in which the BrNN has the appearance of an APPN End Node. The upstream link is likely to be a single WAN link of relatively low bandwidth whereas the downstream network is likely to be a local area network.

More terminology

The term Branch Extender or its abbreviation “BX” refers to the architecture and function as described in:

- *Systems Network Architecture Advanced Peer-to-Peer Networking, Branch Extender Architecture Reference, Version 1.1*, SV40-0129-01.

Branch Network Node or its abbreviation “BrNN” refers to an APPN Network Node that *implements* the APPN Branch Extender Function (also defined by APPN option set 1121).

APPN architecture also defines the Border Node (BN), more specifically the Extended Border Node (EBN) and Peripheral Border Node (PBN). These terms and abbreviations should not be confused with BX/BrNN, form no part of the Cisco APPN implementation, and will not be discussed here.

With the advent of SNASw, Cisco no longer offers a traditional Network Node (other than BrNN, which is technically a special kind of Network Node). Cisco has never offered a pure End Node implementation on its routers.

4.1.1 Summary of BX function

BrNNs can be cascaded as shown in Figure 50 on page 98 and each BrNN represents itself as an EN to upstream devices and as a NN to downstream devices. A significant restriction of cascaded BrNNs is that if DLUR is required, it can only be implemented in the BrNN that connects directly to the network in which the DLUS is located (the first BrNN in the cascaded chain, if you will). This means that BrNNs should never be cascaded in any configuration where support of dependent LU traffic (mainly 3270 screens and printers) downstream of the first BrNN is required; independent LU6.2 traffic is unaffected by this restriction.

Design rationale

The purpose of the BX design stems from the fact that APPN Network Nodes implement and maintain a network topology database; NNs maintain information on the status of links in the network and participate in searches through the network for resources.

From the perspective of a branch network, there is no real need to know about the topology of the entire network, and any resource that isn't located in the branch itself must be located through the central site if it is actually anywhere in the network.

If the branch network contains no downstream APPN devices (no LEN or EN devices, as shown in Figure 50 on page 98) then the node which connects to the central site can be configured as an End Node. It can then be configured to serve dependent SNA devices using DLUR (it is only the DLUS implementation that is required to be implemented on an APPN Network Node).

BX allows the implementation of a device that does not have to maintain a complete topology database (it only has to maintain one for the downstream network of the branch itself) and which does not have to participate in network searches. This allows a simpler APPN implementation and a reduction in the use of wide area bandwidth. But the BrNN does support APPN devices in the branch itself, essentially allowing a full APPN branch implementation without much of the cost overhead of having to implement a full APPN Network Node. Because BrNNs do not participate in network broadcast searches or in network topology updates, the use of BrNNs allows for a much more scalable APPN network. Network reliability and stability is increased because BrNNs are immune from storms of network broadcast searches, which can result from repeated searches for non-existent resources in large SNA networks and can cause network stability problems when these searches are repeatedly sent over low-capacity wide-area links.

The other aspect of the BX architecture is that the BrNN only has a single upstream CP-CP session. Since the BrNN has an appearance of an APPN End Node in this network, this link is to the BrNN's Network Node Server (NNS). The BrNN can define other links to other NNs and ENs in the upstream network, but it will not have CP-CP sessions with any of these APPN nodes. The BrNN can and probably should also define a link to an alternate Network Node to act as a backup NNS. The purpose of this design is so that the BrNN only has a single link in the upstream network over which it can send search requests for unknown resources; the NNS acts as a kind of “default router. Thus the BrNN need only maintain a topology database for the downstream branch network, and simply forwards requests for any other resources over the single upstream link.

4.1.2 Network design considerations

This section reviews some of the network design issues that should be considered by organizations who run large APPN networks today and who may wish to add Cisco routers to the network.

The key point to note is that adding Cisco routers with SNASw function will require a careful insertion strategy if there are currently a number of cascaded NNs in the network path from the downstream device to the hosts. The BX function must not have traditional NNs below it in the network configuration, so you must add Cisco SNASw routers at the bottom level of the APPN network, so to speak. If further removal of network based NNs is desired, the next step is to design for direct links between the BX nodes and the CS/390 hosts, instead of having a complex cascaded NN network in the middle. This is typically done by using Enterprise Extender so that an IP network can connect the BX node directly to the hosts without the need for intermediate NNs. IBM 221x routers can be used if intermediate Network Node function is required in the network, but the simplest configuration to set up and maintain will be one where Cisco SNASw routers can establish links directly with host Network Nodes or End Nodes without the need for intermediate Network Nodes.

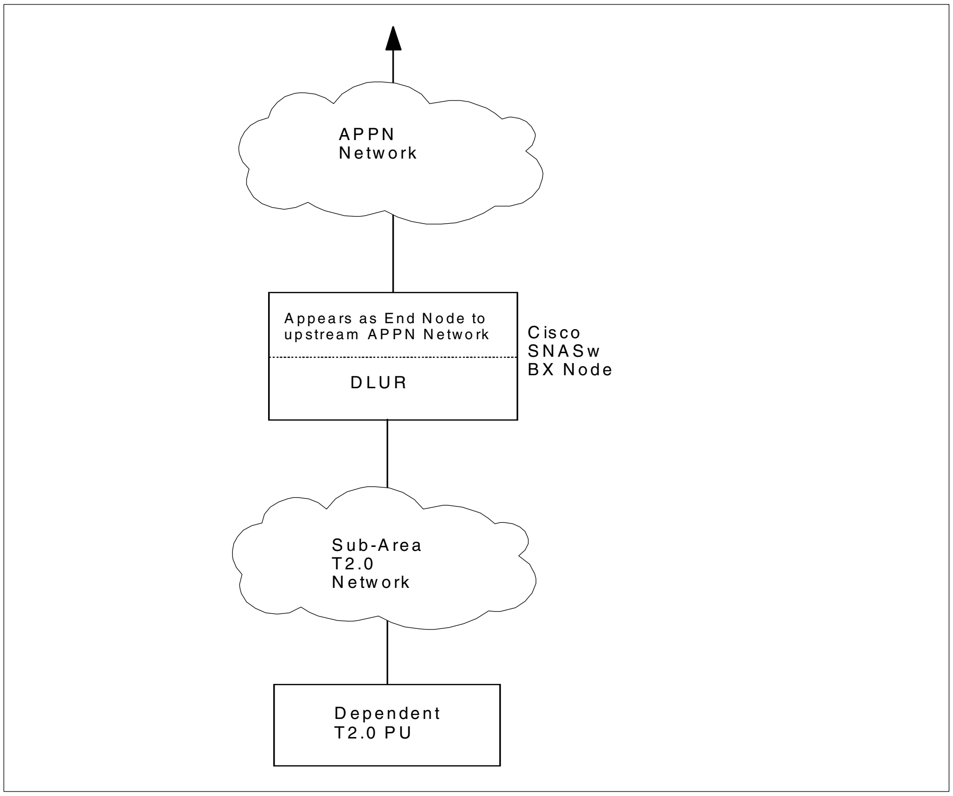


Figure 51. SNASw is the boundary between the APPN and subarea network.

APPN nodes upstream of the Cisco SNASw node in Figure 51 must be full APPN Network Nodes, because DLUR is not supported in cascaded BX nodes. This means that any upstream routers in the network must be IBM 221x routers or similar devices which can be configured as APPN Network Nodes.

4.1.2.1 APPN network design consisting entirely of Cisco SNASw

If a network is to be based entirely on Cisco routers then two basic SNA network designs are possible. The first implements the SNASw function in all branch or remote site routers and is shown in Figure 52. The second restricts the SNASw function to the central site and is shown in Figure 53 on page 106. In both cases the central site includes a VTAM Network Node and can operate as a Parallel Sysplex.

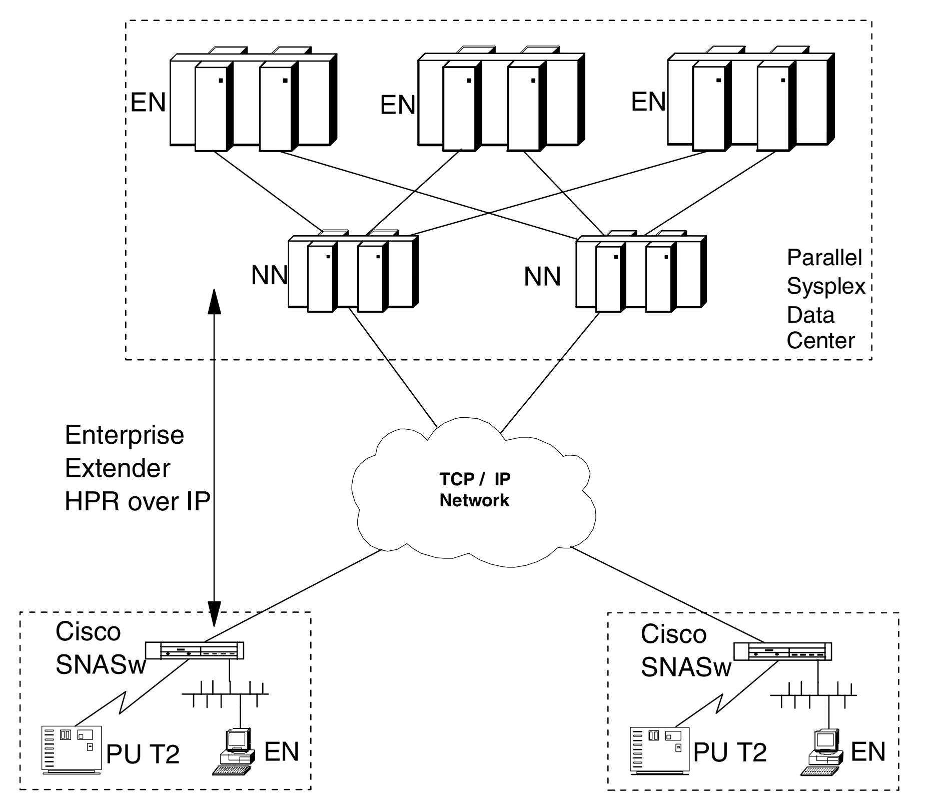


Figure 52. SNASw implementation in branch routers

S/390 Parallel Sysplex

A Parallel Sysplex is an implementation of a clustering and coupling technique for multiple S/390 mainframes that provides an environment for non-disruptive 24 hour/day 7 day/week operation. A Parallel Sysplex *requires* that APPN be implemented in the data center itself, and the benefits of full availability can be extended into the network if APPN is also implemented in the network itself.

In a S/390 mainframe environment that is not configured as a Parallel Sysplex, on the other hand, implementation of APPN in the data center is *optional*, as a configuration option. It is most likely, however, that if a BX network is built using Cisco routers that implement SNASw, then the upstream Network Node Server *will* be a S/390 implementation of APPN.

The first approach (Figure 52 on page 103) allows full APPN HPR connectivity between the data center and remote sites. It supports HPR transport natively over the WAN or via HPR over IP (Enterprise Extender). DLUR may be implemented in the remote site Cisco SNASw nodes to support downstream T2.0 nodes. The core TCP/IP network infrastructure may consist of any routers and the data center mainframes may connect to the TCP/IP infrastructure using channel-attached routers or OSAs (S/390 Open Systems Adapters). The mainframes that connect directly to the IP network need to be running at least either OS/390 V2R6 with PTF OW36113 or OS/390 V2R7.

Enterprise Extender

APPN networks, like all networks, are comprised of nodes and links between the nodes; in APPN's case the links are called transmission groups or TGs. APPN provides for different transmission media types for its TGs, which historically have been thought of either as WAN links (frame relay, PPP) or as LAN links (token-ring, Ethernet). Enterprise Extender introduces a new transport mechanism for APPN: it uses UDP frames and therefore a complete IP network can be viewed as a single TG between APPN nodes. Any IP routers in the core of the network itself do not need to implement APPN since they simply route UDP packets according to their standard IP routing mechanisms.

The reason for referring to Enterprise Extender as "HPR over IP" rather than "APPN over IP" is because it does not support the earlier implementation of APPN - ISR or Intermediate Session Routing.

Figure 52 on page 103 shows a core TCP/IP network and therefore an APPN implementation of Enterprise Extender. In today's world, this is probably the most realistic type of implementation. The Cisco SNASw also supports the more traditional type of APPN network implementation, using LAN and WAN links that support APPN. For example, APPN traffic could be bridged through the network between layer 2 (MAC) addresses of the Branch Network Node and mainframe itself, or it could be transported over frame relay links to a central-site APPN Network Node, which is in turn attached to a mainframe channel. Most of this chapter will discuss the implementation of HPR over IP links upstream of the Cisco Branch Network Node, but other implementations are possible too.

Connection networks

A final discussion of a realistic implementation should also mention that SNASw supports connection networks over IP. In the previous picture, if the mainframe ENs themselves also have a direct connection into the TCP/IP network (unlike actually shown), and since most actual user sessions will terminate in applications running on the ENs in the data center, we really want direct APPN links between the SNASw branch devices and the ENs themselves. Although these can be defined explicitly, a connection network allows a simple definition of a common virtual node to which all ENs and all SNASw devices connect.

In our case, this common virtual node represents the existence of a link into the common IP network. When a session needs to be established between an SNASw node and a central site EN, APPN's topology and routing services component will recognize the existence of this common link and cause a direct Enterprise Extender link to be set up between the two nodes.

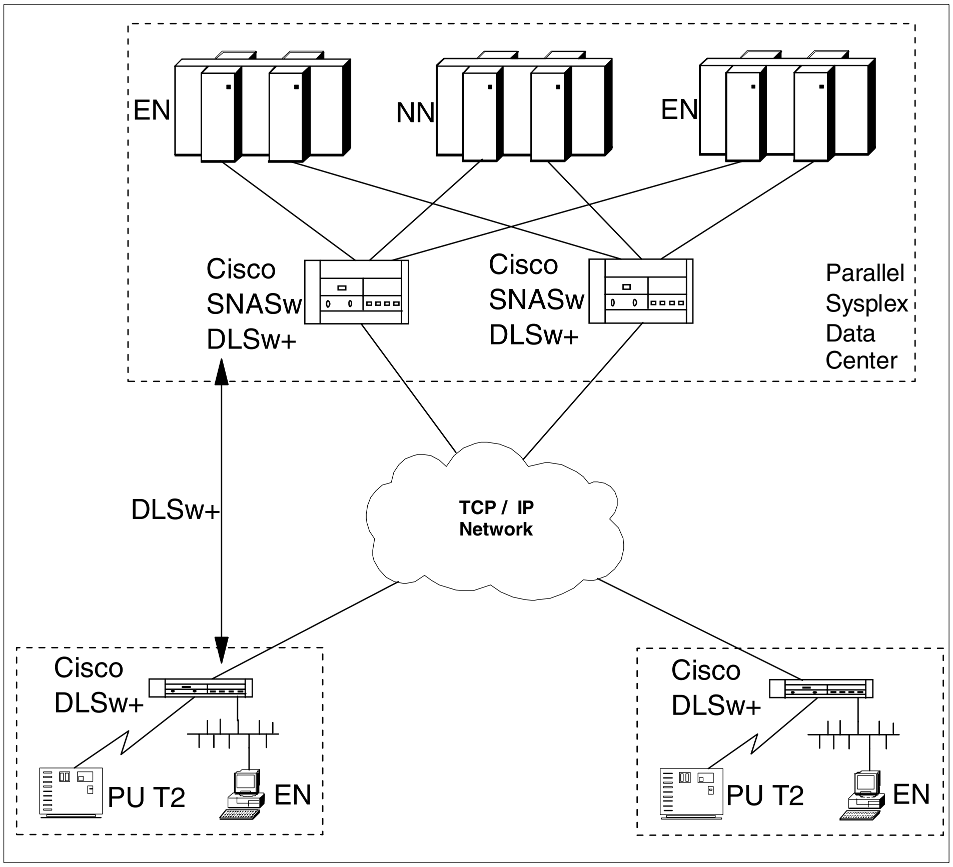


Figure 53. Network using only Cisco routers Implementing SNASw only in the data center

The second approach limits HPR function to the data center itself. A VTAM mainframe Network Node is still required, as shown, to support the Parallel Sysplex, or simply to act as the Cisco BrNNs' Network Node Server if no Parallel Sysplex is configured. SNA traffic must then be carried across the WAN via DLSw or bridging. The data center Cisco SNASw routers then use HPR to connect to the mainframe applications. Dependent SNA sessions will be supported through DLUR implemented in the central SNASw routers. Both dependent and independent LU traffic will be carried over the WAN via DLSw or another encapsulation technique.

Note that the network shown in Figure 53 on page 106 differs from the more "traditional" implementation of SNA support in channel-attached routers in that APPN is now only being used between the channel-attached routers and the mainframe applications. It is likely that this would be implemented for a mainframe Parallel Sysplex environment in which, as has already been mentioned, the mainframe implementation of APPN is a prerequisite anyway; or in cases where CS for OS/390 element address limitations and/or multiple LPAR environments where direct routing of sessions to the target data host from the SNASw router is desired (that is, sessions do not need to be routed through the owning SSCP of an LU to get to the target data host). This design is particularly appealing to networks where a current DLSw infrastructure exists and SNASw/APPN is being used in the data center to replace more traditional subarea networking equipment. For smaller mainframe environments, the existing external communications adapter (XCA) support using link services architecture (LSA) will probably remain the most appropriate implementation model.

4.1.2.2 APPN network design - mixed IBM and Cisco routers

Additional configurations are possible in a network consisting of both IBM and Cisco routers. The most significant additional option allows for a multi-tiered APPN network. In this case the regional hubs are based on IBM routers, such as the 2216, which implements APPN Network Node function as shown in Figure 54 on page 108. It is not possible to build a multi-tiered APPN network using a network of Cisco SNASw nodes if PU T2.0 nodes are to be supported at the edges of the network due to the DLUR restriction of the BX architecture.

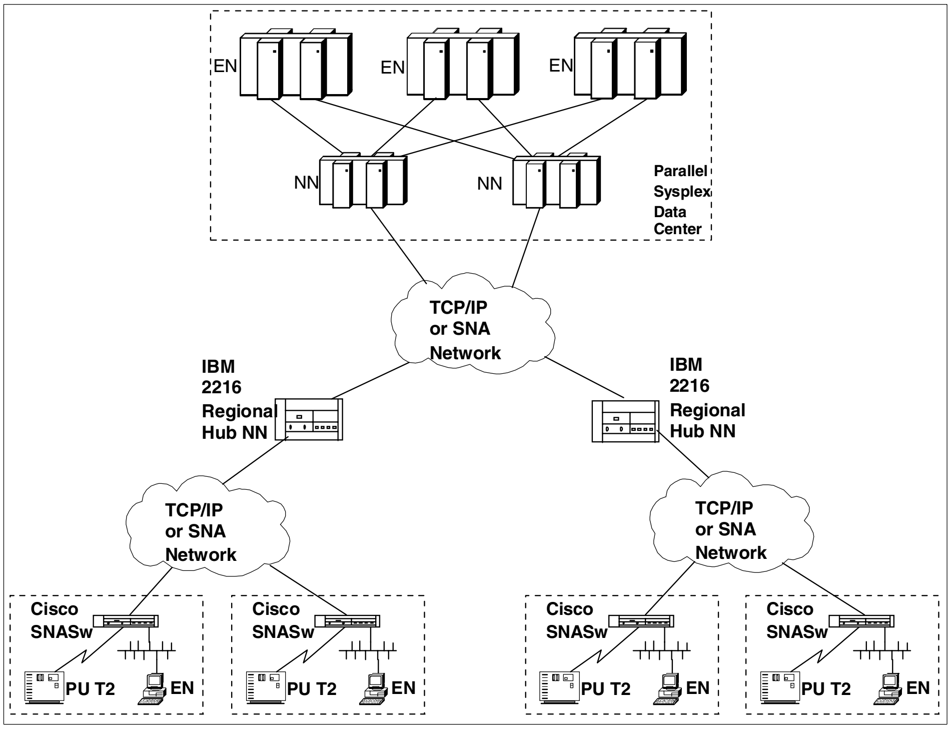


Figure 54. APPN network which includes regional hubs

This diagram shows three networks, any of which may be TCP/IP networks, in which case SNA traffic will be carried via the HPR over IP (Enterprise Extender) function of the Cisco (SNASw) and IBM (NN) routers[2]. If in fact *all* the networks are TCP/IP networks, a better solution will probably be one which eliminates the middle Network Nodes and looks like Figure 52 on page 103 instead. Figure 54 shows what is possible rather than what is desirable.

Interoperability tests between IBM and Cisco routers using SNASw and DLSw have shown a high degree of compatibility, so it is possible to construct a mixed network utilizing IBM and Cisco routers at both the data center and remote sites. An example of this type of mixed network configuration is shown in Figure 55 on page 109.

[2] IBM routers have supported Enterprise Extender since Release V2R2, released in 1997.

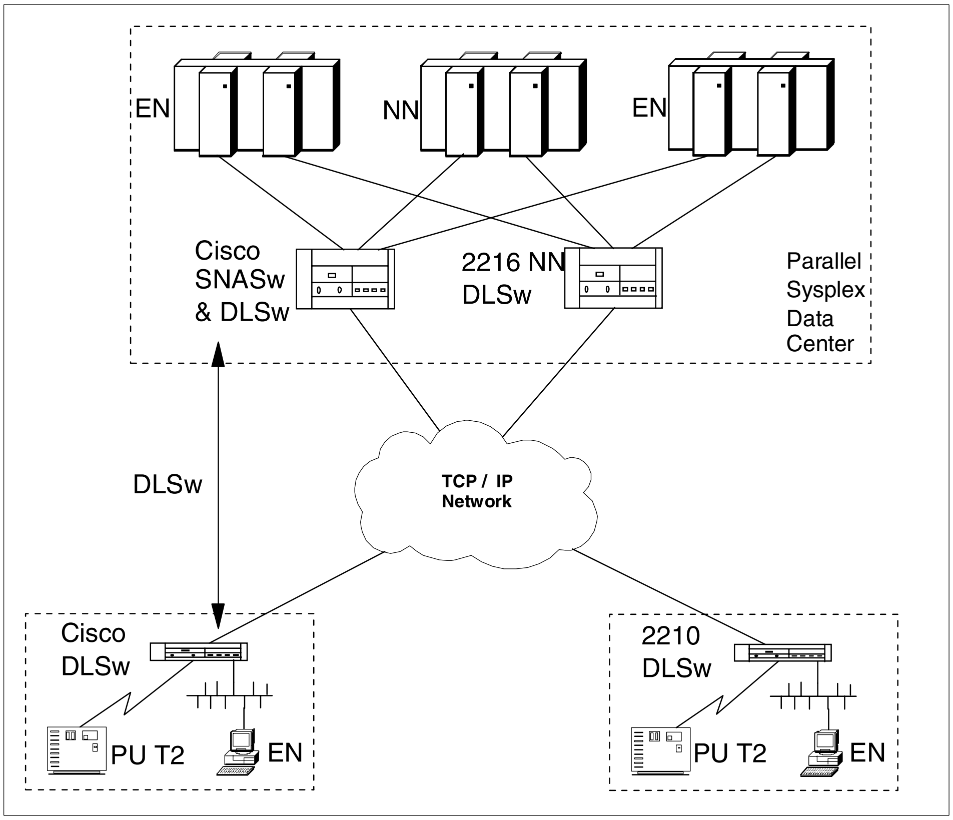


Figure 55. Mixed IBM and Cisco routers in data center and remote sites

In this picture the Cisco routers implement DLSw V2 to provide interoperability with the IBM DLSw function ("DLSw+" is proprietary to Cisco). The data center Cisco and IBM routers both use APPN HPR connectivity to the data center mainframes.

Note that future APPN networks will likely be based more and more on an IP infrastructure, such that multi-tiered APPN networks are unnecessary. To this end, it is recommended that a multi-tiered APPN network only be used as a intermediate step before eventually converging on a single tier, IP infrastructure-based APPN network with BX routers communicating directly with the hosts in the network. This picture shows what is possible with a mixed IBM/Cisco environment today rather than a desirable final network implementation.

4.1.3 Supported Cisco SNASw link types

The SNASw subsystem supports a range of link types to establish SNA connections with upstream and downstream devices. Supported link types and interfaces include:

- Native SNA transport on token-ring, Ethernet and FDDI.
- Virtual token-ring interfaces that support source route bridged connections to local LANs and to channel interface cards such as the CIP (Channel Interface Processor) and CPA (Channel Port Adapter).
- SNA over frame relay using bridged format RFC1490 frames (BAN, or Boundary Access Node).
- DLSw+ links (although it is likely that Enterprise Extender would normally be preferable for the transport of APPN traffic).
- Directly attached SDLC and QLLC links.

SNA over frame relay using routed format RFC1490 Frames (BNN) is *not* supported.

SNASw APPN ports can support both HPR and non-HPR (ISR) traffic. Ports are configured by default to support HPR traffic over LAN and WAN links. HPR can be specifically disabled at the port level.

Any port can also be configured to support HPR over IP (Enterprise Extender) connections via the LAN or WAN TCP/IP infrastructure. SNASw links configured to use these ports specify the remote IP address of the APPN partner node. Any other SNASw link definitions configured on a Cisco router must specify the remote MAC address of the partner APPN node.

4.1.4 Basic configuration commands

Most Cisco SNASw node configurations can be completed using a small number of simple commands.

To enable the SNASw node subsystem and define the node name, we use the `snasw cpname` command as shown below.

```
snasw cpname APPNNET.BONN2600
```

This statement has defined a SNASw node with the fully qualified name APPNNET.BONN2600.

As with IBM routers, interfaces are enabled for APPN by defining them as an APPN port using the `snasw port` statement as shown below.

```
snasw port ETHSNASW FastEthernet0/0
```

This statement has enabled APPN routing on the Fast Ethernet port 0/0.

To establish links between APPN nodes, link stations are defined using the `snasw link` command as shown below.

```
snasw link RAL2216 port FRAME ip-dest 10.1.255.20
```

This statement has defined an Enterprise Extender (HPR over IP) APPN link called RAL2216 via an APPN port called FRAME to a remote SNA node identified by the IP address 10.1.255.20.

The full range of configuration statements are available in documentation on the Cisco Web site (`http://www.cisco.com`).

4.1.5 Interoperability test 1

An APPN network configuration was constructed as shown in Figure 56 on page 112. This configuration involved three IBM routers running both TCP/IP and APPN routing functions. Both Raleigh routers were configured as full APPN Network Nodes.

This configuration was set up to investigate general connectivity between IBM and CISCO APPN implementations. It included HPR over IP, native HPR and non-HPR links and both LAN and WAN connections.

Both Bonn routers were configured as Branch Network Nodes; the Bonn Cisco 2621 used the SNASw function of IOS and the Bonn IBM 2212 used the BX function of IBM’s router common code. This means that the Bonn network shows the use of cascaded Branch Network Nodes.

Cascaded BrNN

In our test environment we chose to implement a cascaded Branch Network Node: the Bonn IBM 2212 cascaded behind the Bonn Cisco 2621. Because of the limitations of this configuration, a more realistic and flexible configuration would have been one in which both devices in the Bonn branch office were configured to use Enterprise Extender and connect directly to the central Network Node: traffic from the Bonn 2212 would then be transported by the IP routing function of the Bonn 2621. Our test network was designed to show that a cascaded BrNN configuration is possible, not that it is necessarily desirable.

The APPN link between the two Raleigh IBM Network Nodes was defined as a non-HPR link simply for variety; a real LAN link in such an environment would normally be defined as supporting HPR whenever possible.

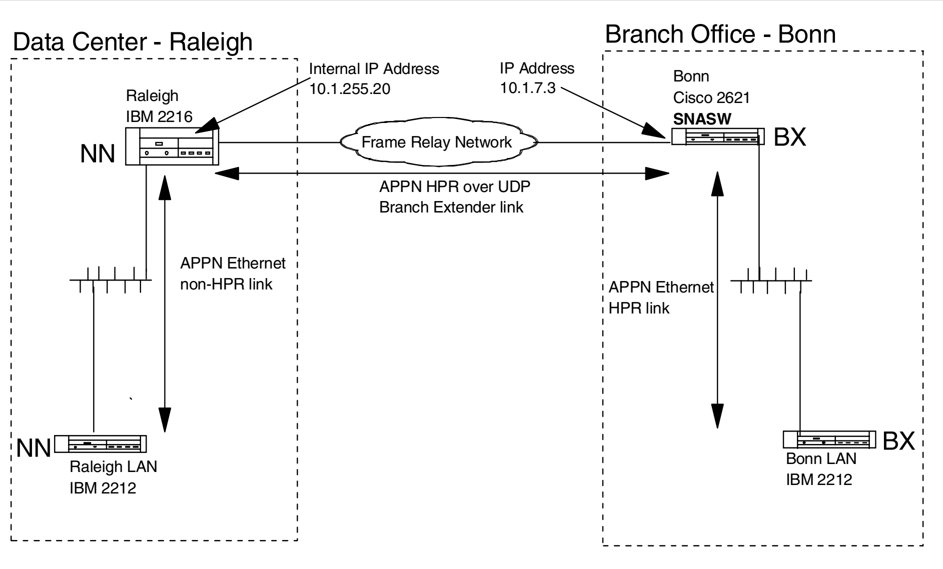


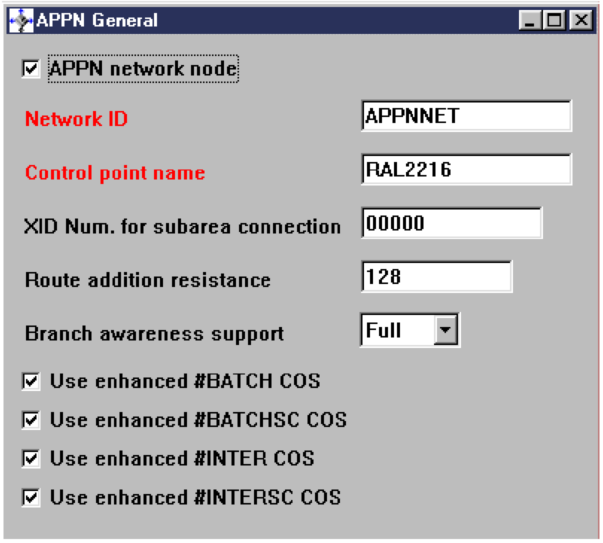
Figure 56. APPN network including Cisco SNASw configuration

Communication between the Raleigh IBM 2216 Network Node and the Bonn Cisco 2600 SNASw node was via Enterprise Extender (HPR over IP).

This configuration allowed APPN connectivity and full SNA communication was achieved between all APPN nodes. The relevant configuration parameters for the nodes are reviewed in the following sections.

4.1.5.1 IBM 2216 router configuration

The Raleigh IBM 2216 is configured as a Network Node as shown in Figure 57 on page 113.



☑ APPN network node

Network ID	APPNNET
Control point name	RAL2216
XID Num. for subarea connection	00000
Route addition resistance	128
Branch awareness support	Full

☑ Use enhanced #BATCH COS

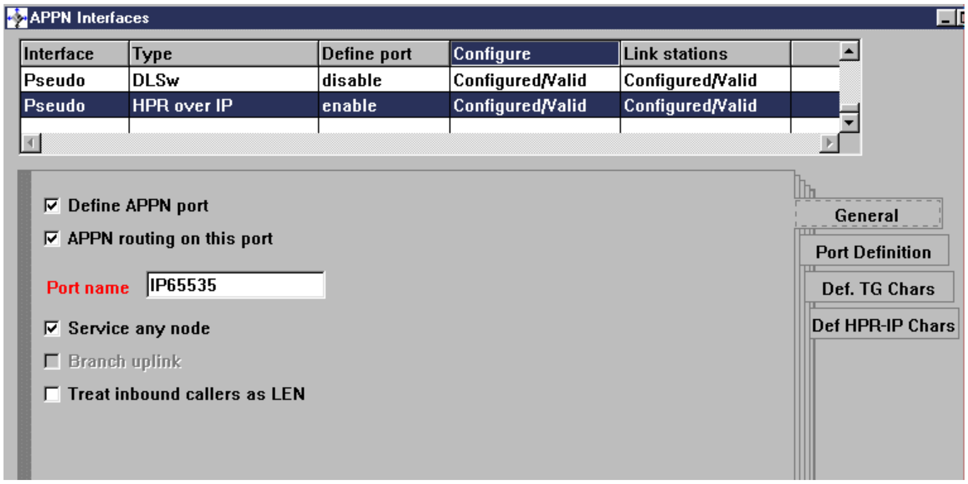
☑ Use enhanced #BATCHSC COS

☑ Use enhanced #INTER COS

☑ Use enhanced #INTERSC COS

Figure 57. IBM 2216 APPN node definition

The APPN HPR over IP port (Enterprise Extender) is enabled in the APPN interfaces panel as shown in Figure 58. This function requires the use of the IBM 2216's internal IP address. It is not necessary to define any link stations as the link will be activated by the remote Bonn Cisco 2621 SNASw node.



Interface	Type	Define port	Configure	Link stations
Pseudo	DLSw	disable	Configured/Valid	Configured/Valid
Pseudo	HPR over IP	enable	Configured/Valid	Configured/Valid

☑ Define APPN port

☑ APPN routing on this port

Port name IP65535

☑ Service any node

☐ Branch uplink

☐ Treat inbound callers as LEN

Figure 58. Enable IBM 2216 HPR over IP - Branch Extender function

APPN is also enabled on the IBM 2216's Fast Ethernet port to allow normal APPN connectivity to local APPN End Nodes such as the Raleigh IBM 2212.

4.1.5.2 Cisco 2621 router configuration

The Bonn Cisco 2621 router configuration statements required to implement the SNASw function are shown in Figure 59.

```
!
interface FastEthernet0/0
 mac-address 0200.2600.1111 1
 ip address 10.1.11.3 255.255.255.0
 no ip directed-broadcast
 duplex auto
 speed auto
!
interface Serial0/0
 mtu 2044
 no ip address
 no ip directed-broadcast
 encapsulation frame-relay IETF 2
 no ip mroute-cache
 no arp frame-relay
 frame-relay traffic-shaping
 frame-relay lmi-type ansi
!
interface Serial0/0.1 multipoint 3
 ip address 10.1.7.3 255.255.255.0 4
 no ip directed-broadcast
 ip ospf network point-to-multipoint
 ip ospf hello-interval 10
 no arp frame-relay
 frame-relay interface-dlci 272 5
!
!
snasw cpname APPNNET.BONN2600 6
snasw port ETHSNASw FastEthernet0/0 7
snasw port FRAME hpr-ip Serial0/0.1 8
snasw link RAL2216 port FRAME ip-dest 10.1.255.20 9
!
```

Figure 59. Bonn Cisco 2621 router configuration for SNASw functions

The significant points in the Bonn Cisco 2621 router configuration are:

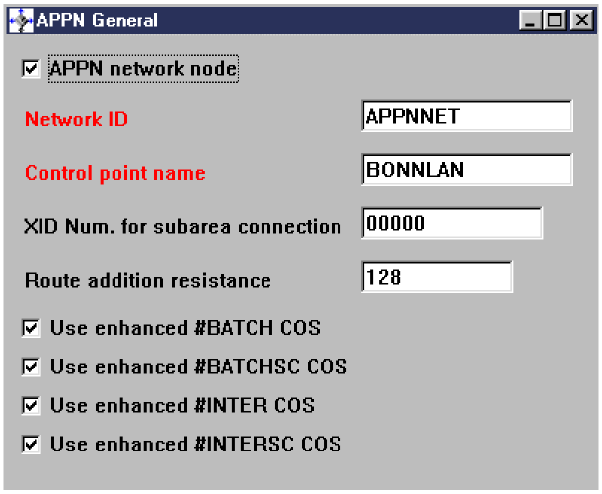
- A locally administered MAC address (LAA) is assigned to the Ethernet interface to support APPN communication with the downstream Bonn IBM 2212 Branch Network Node. 1
- Frame relay encapsulation is set to RFC1490/IETF to be compatible with the IBM 2216. 2

- The serial port subinterface is defined to allow communication with the central IBM 2216 using both IP and APPN. 3
- An IP address is assigned to the frame relay subinterface for normal TCP/IP communication and APPN HPR over IP. 4
- The frame relay subinterface is linked to a particular frame relay PVC. 5
- The Bonn Cisco 2621’s APPN fully qualified node name is set to APPNNET.BONN2600. 6
- An APPN port is enabled on the Fast Ethernet interface. 7
- An APPN port we called “FRAME” is enabled on the frame relay subinterface for HPR over IP communication with the central IBM 2216. 8
- A link station is defined on the FRAME port to point to the central IBM 2216’s internal IP address. 9

Note: Defining the HPR over IP link station to point to the IBM 2216’s frame relay interface IP address does not work. HPR over IP on the IBM 2216 uses the 2216’s internal IP address.

4.1.5.3 Bonn IBM 2212 Branch Network Node configuration

The Bonn IBM 2212 router is configured as a Branch Network Node downstream from the Cisco 2621 SNASw BrNN. First the IBM 2212’s APPN fully qualified node name is set as shown in Figure 60.



☑ APPN network node

Network ID APPNNET

Control point name BONNLAN

XID Num. for subarea connection 00000

Route addition resistance 128

☑ Use enhanced #BATCH COS

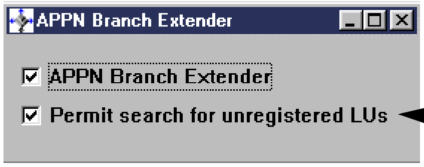
☑ Use enhanced #BATCHSC COS

☑ Use enhanced #INTER COS

☑ Use enhanced #INTERSC COS

Figure 60. Bonn IBM 2212 APPN node definition

Branch Extender function is then enabled on the Bonn IBM 2212 as shown in Figure 61 to allow it to appear as an End Node to the Cisco 2621. The Cisco 2621 will act as the IBM 2212’s Network Node Server in this environment, just as it would for any other SNA nodes on the Bonn LAN.



APPN Branch Extender

☑ APPN Branch Extender

☑ Permit search for unregistered LUs

We should not have done this!

Figure 61. Enable BX function on Bonn IBM 2212

Searching BrNNs

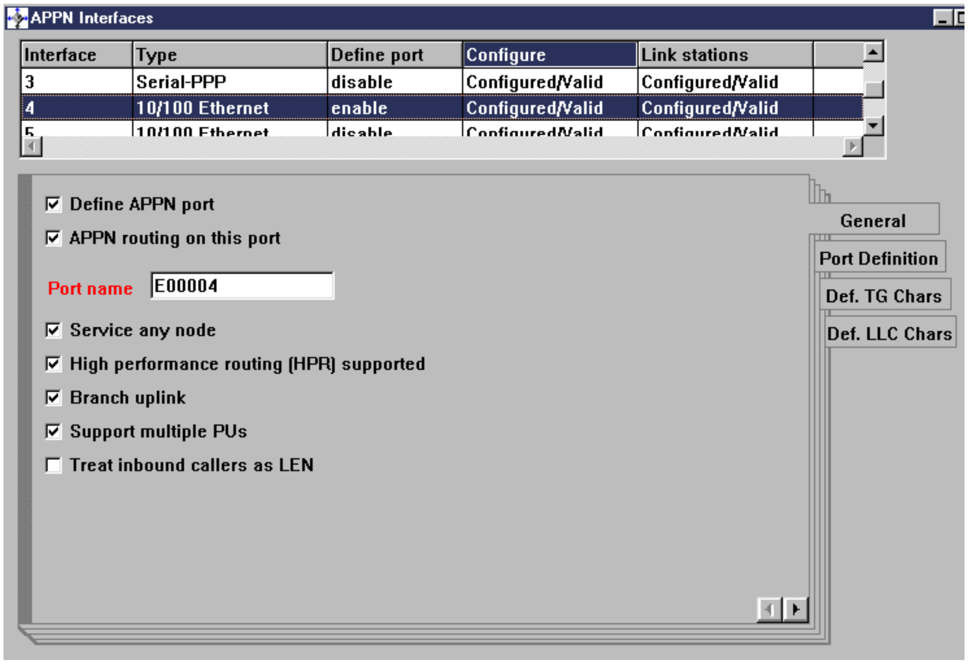
The negative impact of allowing our BrNN to be searched would not have been noticed in our small network, but this is not something that should be enabled as a matter of course in a real network.

If all the real APPN End Nodes in the branch network behind the BrNN register all their LUs with the BrNN acting as their Network Node Server, then the BrNN in turn registers all the LUs with *its* NNS. This means that all the branch LUs will always be known by the BrNN’s NNS, so there is no point attempting to search the BrNN for other resources.

More complex SNA networks than ours often have a problem in that repeated searches for non-existent resources take place. These searches are triggered by incorrect or out-of-date partner LU definitions in network devices. APPN Network Nodes do not save information on non-existent resources, so every time a device attempts to locate a non-existent resource it will cause a broadcast search of the entire network. We really should prevent these searches from flowing to BrNNs over low-capacity WAN links whenever possible. They should only be allowed if we need to cater for the existence of LUs in the branch network that do not register their existence with the Branch Network Node, and which therefore will not automatically be known to the rest of the network.

Indeed, although allowing this search option is defined in the BX architecture, Cisco appears to go one step further and does not allow configuration of this option in its SNASw implementation. This is probably wise, although it precludes the existence of LUs that are not automatically registered or manually predefined in the branch network itself.

The Ethernet interface on the Bonn IBM 2212 is then enabled for APPN and is defined as a branch uplink interface as shown in Figure 62, to allow it to appear as an End Node to the Cisco 2621.



Interface	Type	Define port	Configure	Link stations
3	Serial-PPP	disable	Configured/Valid	Configured/Valid
4	10/100 Ethernet	enable	Configured/Valid	Configured/Valid
5	10/100 Ethernet	disable	Configured/Valid	Configured/Valid

☑ Define APPN port

☑ APPN routing on this port

Port name E00004

☑ Service any node

☑ High performance routing (HPR) supported

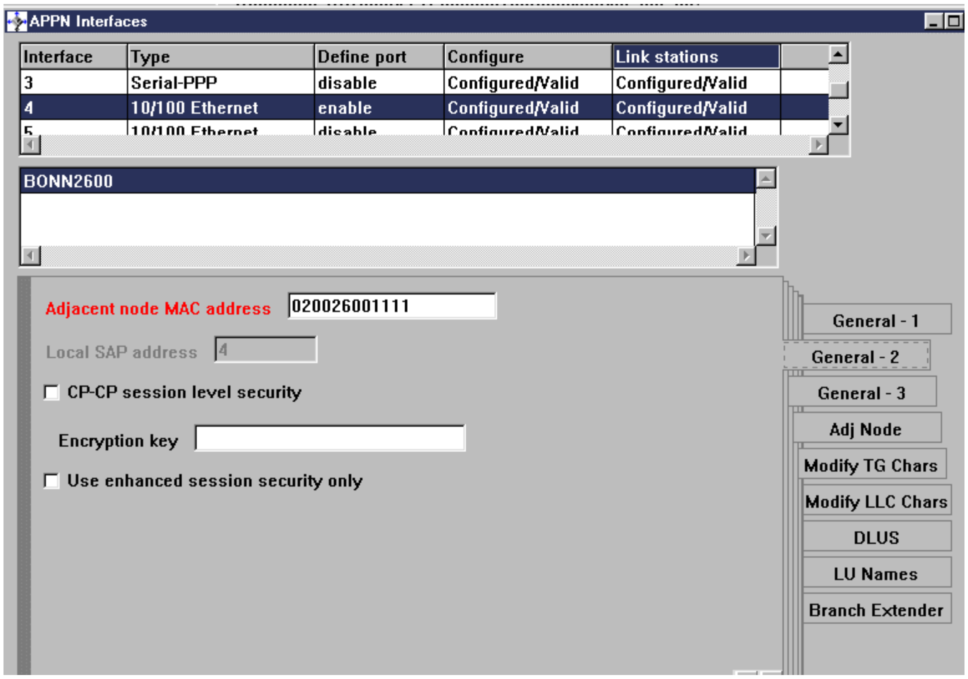
☑ Branch uplink

☑ Support multiple PUs

☐ Treat inbound callers as LEN

Figure 62. Enable Ethernet interface as APPN port and set to branch uplink

The APPN link to the upstream Cisco 2621 BrNN is then defined as shown in Figure 63, specifying the LAA of the upstream Cisco 2621.



Interface	Type	Define port	Configure	Link stations
3	Serial-PPP	disable	Configured/Valid	Configured/Valid
4	10/100 Ethernet	enable	Configured/Valid	Configured/Valid
5	10/100 Ethernet	disable	Configured/Valid	Configured/Valid

BONN2600

Adjacent node MAC address 020026001111

Local SAP address 4

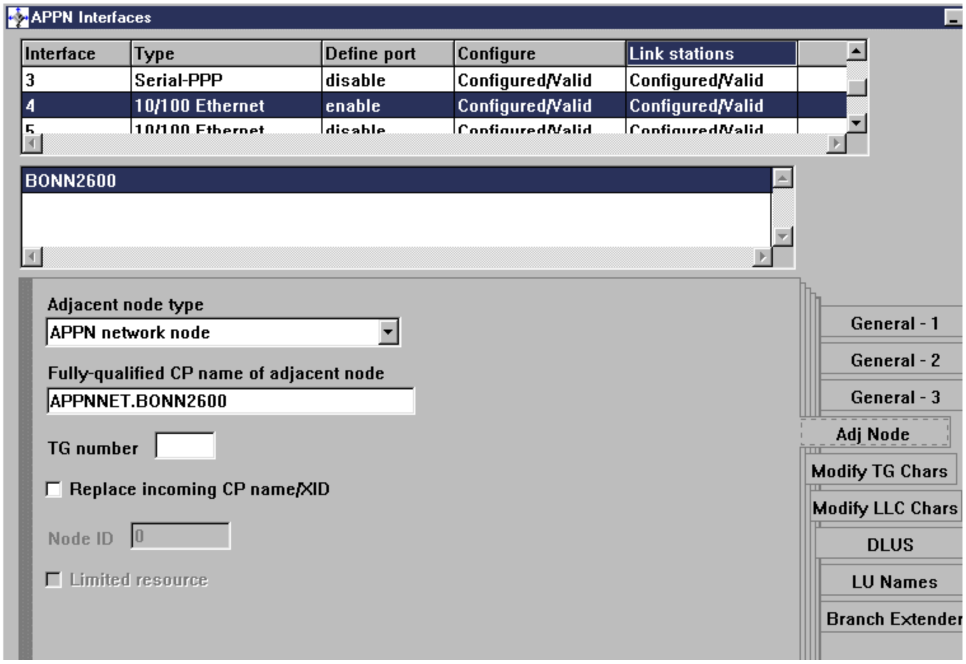
□ CP-CP session level security

Encryption key

□ Use enhanced session security only

Figure 63. Bonn IBM 2212 link station to Cisco 2621 Ethernet MAC address is specified

The Upstream Cisco 2621 BrNN is defined to the Bonn IBM 2212 as a Network Node, because this is how it will appear to the Bonn IBM 2212 LAN router. This definition is shown in Figure 64.



Interface	Type	Define port	Configure	Link stations
3	Serial-PPP	disable	Configured/Valid	Configured/Valid
4	10/100 Ethernet	enable	Configured/Valid	Configured/Valid
5	10/100 Ethernet	disable	Configured/Valid	Configured/Valid

BONN2600

Adjacent node type: APPN network node

Fully-qualified CP name of adjacent node: APPNNET.BONN2600

TG number

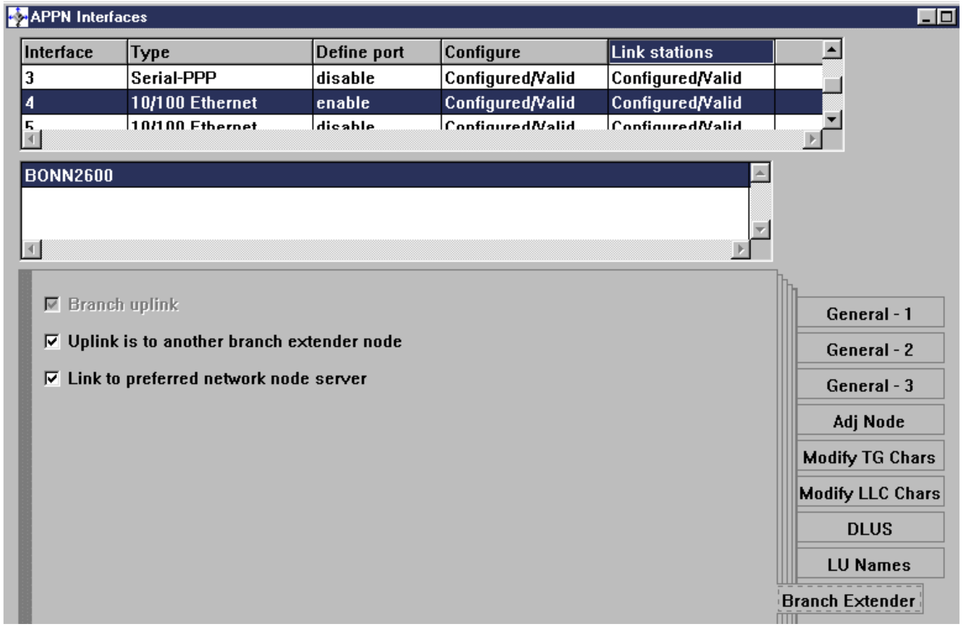
☐ Replace incoming CP name/XID

Node ID 0

☐ Limited resource

Figure 64. Define link to Cisco 2621 as Network Node as it is the upstream BX node

Finally the Bonn Cisco 2621 BrNN is defined as the preferred Network Node Server and also as a link to another BrNN as shown in Figure 65.



Interface	Type	Define port	Configure	Link stations
3	Serial-PPP	disable	Configured/Valid	Configured/Valid
4	10/100 Ethernet	enable	Configured/Valid	Configured/Valid
5	10/100 Ethernet	disable	Configured/Valid	Configured/Valid

BONN2600

☑ Branch uplink

☑ Uplink is to another branch extender node

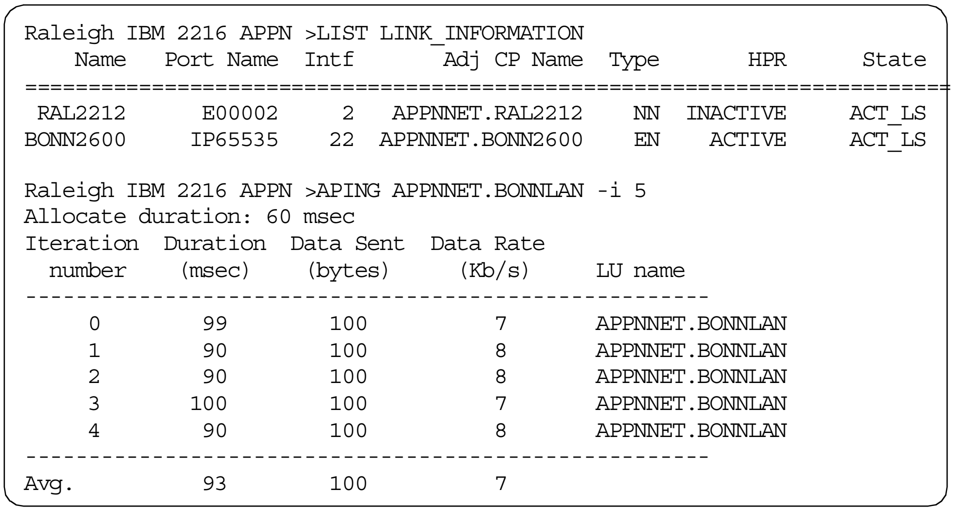
☑ Link to preferred network node server

Figure 65. Define link to Bonn Cisco 2621 as the uplink to the preferred NNS

This completes the configuration of APPN on the Bonn LAN 2212.

4.1.5.4 Test results

Full APPN connectivity was established between the four devices shown in Figure 56 on page 112. The APPN links active on the Raleigh IBM 2216 are shown in Figure 66.



```
Raleigh IBM 2216 APPN >LIST LINK_INFORMATION
    Name   Port Name  Intf       Adj CP Name  Type       HPR      State
=========================================================================
 RAL2212      E00002     2   APPNNET.RAL2212    NN  INACTIVE     ACT_LS
BONN2600     IP65535    22  APPNNET.BONN2600    EN    ACTIVE     ACT_LS

Raleigh IBM 2216 APPN >APING APPNNET.BONNLAN -i 5
Allocate duration: 60 msec
Iteration  Duration  Data Sent  Data Rate
  number    (msec)    (bytes)     (Kb/s)     LU name
-------------------------------------------------------
     0        99        100          7       APPNNET.BONNLAN
     1        90        100          8       APPNNET.BONNLAN
     2        90        100          8       APPNNET.BONNLAN
     3       100        100          7       APPNNET.BONNLAN
     4        90        100          8       APPNNET.BONNLAN
-------------------------------------------------------
Avg.          93        100          7
```

Figure 66. Raleigh IBM 2216 Network Node links and connectivity

Links to both the IBM 2212 on the Raleigh LAN via the Ethernet APPN port and the Bonn Cisco 2621 SNASw BrNN are active. The LAN link to the Raleigh IBM 2212 was configured as a non-HPR link whereas the HPR over IP link to the Bonn Cisco 2621 obviously has HPR active. The results of the APING to the downstream Bonn IBM 2212 LAN router via the Cisco 2621 BX node are also shown.

The `show snasw link` command is used on the Cisco 2621 SNASw node to show the state of APPN links as shown in Figure 67 on page 122. The Cisco 2621 SNASw node sees the upstream Raleigh IBM 2216 as a Network Node via the HPR over IP link. It sees the downstream Bonn LAN IBM 2212 as an End Node. HPR is active on both links. APING is performed on the Cisco routers using the `ping SNA` command as shown in Figure 67. Only the aggregate result is displayed rather than returning a line for each APING, as is the case with IBM routers.

```
Bonn-Cisco#show snasw link
Number of links 2

     SNA Links                                                          HPR
     Link Name   State    Port Name Adjacent CP Name  Node Type    Sess  Sup
     --------- --------  --------- ----------------  ------------ ----  ---
  1> @I000002   Active    ETHSNASW  APPNNET.BONNLAN   End Node        0  Yes
  2> RAL2216    Active    FRAME     APPNNET.RAL2216   Network Node    0  Yes

Bonn-Cisco#ping sna -i 10 APPNNET.RAL2212
Bonn-Cisco#
SNA APING successful
Partner LU name            APPNNET.RAL2212
Mode name                  #INTER
Allocate duration          4 ms
Duration statistics        Min = 92 ms   Ave = 172 ms   Max = 616 ms
```

Figure 67. Bonn Cisco 2621 SNASw BrNN APPN links

The active APPN links visible to the IBM 2212 on the Bonn LAN are shown in Figure 68. The only APPN link is to the upstream Cisco 2621 BrNN, which the IBM 2212 sees as a Network Node.

```
Bonn 2212 APPN >list link
    Name   Port Name  Intf       Adj CP Name  Type       HPR      State
=======================================================================
BONN2600      E00004     4  APPNNET.BONN2600    NN    ACTIVE     ACT_LS
Bonn 2212 APPN >aping APPNNET.RAL2212 -i 5
Allocate duration: 0 msec
Iteration  Duration  Data Sent  Data Rate
  number    (msec)    (bytes)     (Kb/s)     LU name
------------------------------------------------------
     0       100        100          7       APPNNET.RAL2212
     1        90        100          8       APPNNET.RAL2212
     2       379        100          2       APPNNET.RAL2212
     3        90        100          8       APPNNET.RAL2212
     4        90        100          8       APPNNET.RAL2212
------------------------------------------------------
Avg.         149        100          6
```

Figure 68. Bonn LAN IBM 2212 BrNN links and connectivity

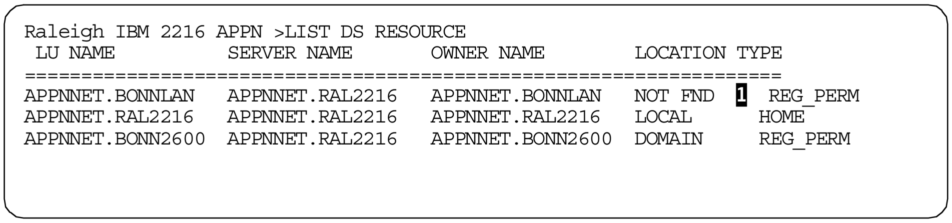
APPN end-to-end connectivity between the remote LAN-connected IBM 2212s via the Cisco SNASw Node 2621 and the Raleigh IBM 2216 is successful as shown by the APING in Figure 68.

Finally the APPN links active on the IBM 2212 on the Raleigh LAN are shown in Figure 69. The only link is to the Raleigh IBM 2216. This link was configured as non-HPR. The APING via this link through to the IBM 2212 BrNN on the Bonn LAN was successful. This APING traversed both non-HPR links and HPR links.

```
Raleigh LAN 2212 APPN >LIST LINK_INFORMATION
    Name   Port Name  Intf       Adj CP Name  Type       HPR      State
=========================================================================
     @@5      E00005     5   APPNNET.RAL2216    NN  INACTIVE     ACT_LS
Raleigh LAN 2212 APPN >APING APPNNET.BONNLAN -i 100
Allocate duration: 60 msec
Iteration  Duration  Data Sent  Data Rate
  number    (msec)    (bytes)     (Kb/s)     LU name
------------------------------------------------------
Raleigh LAN 2212 APPN >APING APPNNET.BONNLAN -i 5
Allocate duration: 70 msec
Iteration  Duration  Data Sent  Data Rate
  number    (msec)    (bytes)     (Kb/s)     LU name
------------------------------------------------------
     0       120        100          6       APPNNET.BONNLAN
     1       110        100          7       APPNNET.BONNLAN
     2       110        100          7       APPNNET.BONNLAN
     3       110        100          7       APPNNET.BONNLAN
     4       110        100          7       APPNNET.BONNLAN
------------------------------------------------------
Avg.         112        100          6
```

Figure 69. Raleigh LAN IBM 2212 NN links and connectivity

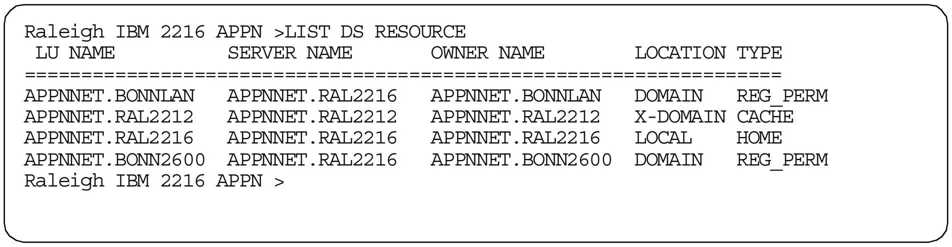
An apparent slight interoperability problem exists between the IBM APPN Network Node and Cisco SNASw node implementations, which initially affected connectivity between the nodes in our test network. As a Branch Network Node, the Cisco 2621 is responsible for registering information about downstream nodes with its upstream Network Node Server, in this case the Raleigh IBM 2216 Network Node (see Figure 56 on page 112). This registration was not initially successful. The IBM 2216 did not initially contain complete information on the location of the Bonn LAN IBM 2212, as shown in Figure 70 on page 124, where the location is listed as `NOT FOUND` **1**. At this point the Raleigh IBM 2216 and IBM 2212 were not able to APING the Bonn LAN IBM 2212.



```
Raleigh IBM 2216 APPN >LIST DS RESOURCE
 LU NAME          SERVER NAME       OWNER NAME        LOCATION TYPE
=================================================================
APPNNET.BONNLAN   APPNNET.RAL2216   APPNNET.BONNLAN   NOT FND  1  REG_PERM
APPNNET.RAL2216   APPNNET.RAL2216   APPNNET.RAL2216   LOCAL       HOME
APPNNET.BONN2600  APPNNET.RAL2216   APPNNET.BONN2600  DOMAIN      REG_PERM
```

Figure 70. Incomplete directory entry for BONNLAN at the Raleigh IBM 2216 NN

This problem seemed to be resolved as soon as the Bonn LAN IBM 2212 itself attempted to communicate with either of the Raleigh nodes, in our case by attempting to use APING. The directory entry in the 2216 NN then changed and full two-way connectivity was then possible. The correct directory entries at the Raleigh IBM 2216 are shown in Figure 71.



```
Raleigh IBM 2216 APPN >LIST DS RESOURCE
 LU NAME          SERVER NAME       OWNER NAME        LOCATION TYPE
=================================================================
APPNNET.BONNLAN   APPNNET.RAL2216   APPNNET.BONNLAN   DOMAIN   REG_PERM
APPNNET.RAL2212   APPNNET.RAL2212   APPNNET.RAL2212   X-DOMAIN CACHE
APPNNET.RAL2216   APPNNET.RAL2216   APPNNET.RAL2216   LOCAL    HOME
APPNNET.BONN2600  APPNNET.RAL2216   APPNNET.BONN2600  DOMAIN   REG_PERM
Raleigh IBM 2216 APPN >
```

Figure 71. Correct directory entry at Raleigh IBM 2216 NN

We did not have time to pursue this apparent problem further before completion of this book, although we did temporarily replace the Cisco 2621 router SNASw node with an IBM 2212 router configured in an identical manner to the Cisco 2621; this configuration did not experience the initial connectivity problems of the mixed configuration and all directory entries were complete and correct at startup.

4.1.6 Interoperability test 2

Our second test was designed to test the DLUR function of the Cisco SNASw software. The test configuration diagram is shown in Figure 72.

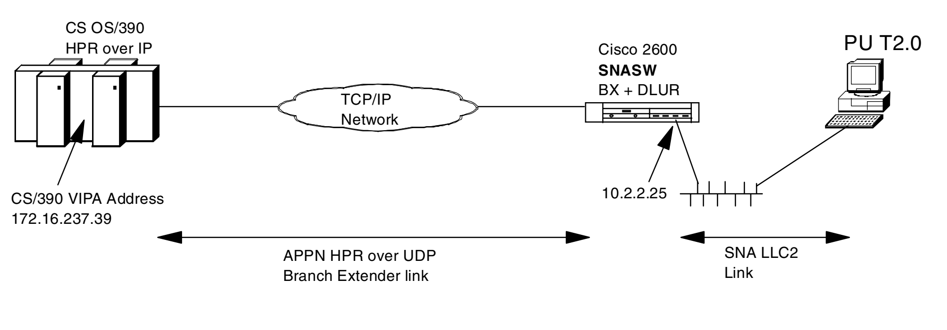


Figure 72. SNASw software and DLUR

CS for OS/390 was configured to support HPR over IP (Enterprise Extender) using the mainframe's own TCP/IP stack and was connected to the TCP/IP network with a S/390 Open Systems Adapter (OSA), which allowed direct TCP/IP connectivity to the Cisco 2600 router running SNASw[3]. The Cisco 2600 SNASw software communicated with the downstream workstation using LLC2 over an Ethernet LAN; the workstation used IBM Personal Communications software as its 3270 emulation software and was configured to use the Cisco 2600 router as its SNA gateway.

The relevant sections of the Cisco router's SNASw configuration are shown in Figure 73.

```
!
interface FastEthernet0/0
 mac-address 0200.2600.1111 1
 ip address 10.2.2.25 255.255.255.0 2
 no ip directed-broadcast
 duplex auto
 speed auto
!
snasw cpname USIBMRA.BONN2600 3
snasw dlus USIBMRA.RA39M 4
snasw port SNASwETH FastEthernet0/0 5
snasw port HPRIPETH hpr-ip FastEthernet0/0 6
snasw link HOSTIP port HPRIPETH ip-dest 172.16.232.39 7
!
```

Figure 73. Cisco 2600 SNASw configuration statements

[3] As it happened, our TCP/IP network comprised multiple token-ring, Ethernet and even ATM segments.

The significant features of this configuration are:

- The locally administered MAC address (LAA) is assigned to the Cisco 2600’s Ethernet port. This is the destination MAC address that is configured in downstream SNA nodes. 1
- The IP address assigned to the Cisco 2600 Ethernet interface. This interface is one end of the IP connection used to transport HPR over IP to the mainframe (Enterprise Extender). 2
- The APPN Control Point name assigned to the Cisco 2600. 3
- The dependent LU server name is configured in the Cisco router: this is the APPN CP name of the VTAM DLUS. 4
- The Ethernet port is configured as an SNASw port for SNA LAN traffic from downstream SNA workstations. 5
- The same Ethernet port is configured to support HPR over IP traffic, because we are in fact using the same physical interface for both upstream and downstream communication. 6
- An HPR over IP APPN link is defined with the TCP/IP address of the mainframe to support HPR over IP traffic. 7

The commands to display the active APPN ports and links on the Cisco 2600 are shown in Figure 74; both the downstream SNA LLC link and the upstream HPR over IP link use the same physical Ethernet interface in this case.

```
Bonn-Cisco#show snasw port
Number of ports 2

      SNA Ports              HPR
        Name     State    SAP  SAP  Interface              Address
      --------  --------  ---  ---  ---------------------  --------------
   1> HPRIPETH  Active              FastEthernet0/0        192.168.140.114
   2> SNASwETH  Active    x04  xC8  FastEthernet0/0        0200.2600.1111

Bonn-Cisco#show snasw link
Number of links 2

      SNA Links                                                        HPR
      Link Name   State    Port Name Adjacent CP Name  Node Type     Sess  Sup
      ---------  --------  --------- ----------------  ------------  ----  ---
   1> @I000001   Active    SNASwETH  USIBMRA.@P000001  PU 2.0           4  No
   2> HOSTIP     Active    HPRIPETH  USIBMRA.RA39M     Network Node     0  Yes
```

Figure 74. Display SNASw ports and links on Cisco 2600 router

The commands shown in Figure 75 display the status of the link from the router to the VTAM DLUS as well as the status of any active downstream PUs and LUs that are using the DLUR function of the SNASw software.

```
Bonn-Cisco#show snasw dlus
Number of Dependent LU Servers 1

      SNA Dependent LU Servers
          DLUS Name      Default?  Backup?  Pipe State         PUs
      -----------------  --------  -------  ----------------  -------
   1> USIBMRA.RA39M      Yes       No       Active                 1

Bonn-Cisco#show snasw lu
Number of DLUR LUs 2

      SNA DLUR LUs
      LU Name   PU Name   DLUS Name          PLU Name
      --------  --------  -----------------  -----------------
   1> EXPL0031  PUDLU3    USIBMRA.RA39M      USIBMRA.RA39T07
Bonn-Cisco#show snasw pu
Number of DLUR PUs 1

      SNA DLUR PUs
      PU Name    PU ID     State       DLUS Name
      --------  --------  --------  -----------------
   1> PUDLU3    05DA642C  Active    USIBMRA.RA39M
```

Figure 75. Display DLUS and DLUR information on Cisco SNASw

The PU’s IDNUM and IDBLOCK are shown in Figure 75 (05D/A642C), as is the LU Name for the dependent LU (RA39T07). These displays are very similar to the equivalent displays on IBM routers.

For completeness, the mainframe VTAM and TCP/IP definitions required to support this configuration are shown in the following figures.

The connection between VTAM and TCP/IP on the mainframe is via an internal XCA interface which specifies “HPRIP” as the medium, as shown in Figure 76.

```
EEXCA    VBUILD TYPE=XCA
EEXCAP   PORT  MEDIUM=HPRIP,SAPADDR=4
EEXCAG   GROUP DIAL=YES
EEXCAL1  LINE  CALL=INOUT
EEXCAP1  PU
EEXCAL2  LINE  CALL=INOUT
EEXCAP2  PU
EEXCAL3  LINE  CALL=INOUT
EEXCAP3  PU
EEXCAL4  LINE  CALL=INOUT
EEXCAP4  PU
```

Figure 76. VTAM XCA definition for HPR over IP

The keyword MEDIUM=HPRIP tells VTAM that this is an Enterprise Extender connection rather than a LAN or ATM connection. The LINE definitions specify DIAL=YES, since Enterprise Extender connections use switched protocols between APPN nodes; this allows VTAM to initiate and activate the connection to the Cisco BrNN by “dialing out” to it across the IP network.

The switched node definitions representing the Cisco 2600 SNASw APPN node itself and the dependent LUs that reside in the PC workstation are shown in Figure 77 and Figure 78. The switched node definition for the Cisco SNASw node includes a PATH statement allowing VTAM to connect out to the Cisco SNASw using the given IP address. This is optional and is not required if the network is configured so that the Cisco BrNN will always initiate an in-bound connection to VTAM. By configuring it, we were pleased to see that the Cisco BrNN became active as soon as the VTAM definitions were activated.

```
EESWJ   VBUILD TYPE=SWNET
EEPU    PU     ISTATUS=ACTIVE,PUTYPE=2,CPNAME=BONN2600
EESW    PATH   IPADDR=192.168.140.114,GRPNM=EEXCAG
```

Figure 77. VTAM switched node definition for Cisco SNASw node

```
SWJACK   VBUILD TYPE=SWNET
PUDLU3   PU    ADDR=02,        WAS ADDR=01                          X
               IDBLK=05D,                                           X
               IDNUM=A642C,                                         X
               PUTYPE=2,                                            X
               CONNTYPE=APPN,                                       X
               USSTAB=US327X,                                       X
               DLOGMOD=D4C32XX3
EXPL0030 LU    LOCADDR=00
EXPL0031 LU    LOCADDR=02,LOGAPPL=RA39T
EXPL0032 LU    LOCADDR=03,LOGAPPL=RA39T
```

Figure 78. VTAM switched node definition for the dependent LUs

Sections of the MVS TCP/IP profile are shown in Figure 79, which include the virtual IP address (VIPA), which the Cisco SNASw link definition points to, and the Enterprise Extender definitions that support the HPR over IP link.

```
IPCONFig
SOURCEVIPA 1
VARSUBNETTING          ; For RIPV2

; *********************************************************
; LCS Definition
; osa ch D8                         Device # 2060-2061
; *********************************************************
 DEVICE TR1    LCS  2060 autorestart
 LINK  TR1    IBMTR    0 TR1i 2
; *********************************************************
; VIPA Definition  (For V2R5)
; *********************************************************
;
   DEVICE VIPA39A  VIRTUAL     0
   LINK   VIPA39A  VIRTUAL     0     VIPA39A 3

;-----------------------------------------------------------------------
;Enterprise extender definition
;-----------------------------------------------------------------------
 DEVICE IUTSAMEH MPCPTP
 LINK   IUTSAMEH MPCPTP IUTSAMEH 4

; HOME Internet (IP) addresses of each link in the host.
HOME
    172.16.232.39 VIPA39A     ; VIPA 5
    9.24.104.149   TR1        ; osa TO public network
    172.16.220.52 loopback    ; ndr cluster
    172.16.223.40 IUTSAMEH    ; EE 6

; Start all the defined devices.
 START TR1                 ; OSA
 START EN1                 ; Ethernet
 START IUTSAMEH            ; EExtender
```

Figure 79. OS/390 TCP/IP profile statements relating to Enterprise Extender

The key configuration components are:

- SOURCEVIPA tells TCP/IP to return the VIPA address as the source address on all outbound datagrams. It also means that a client program on this MVS image (in our case, VTAM is that client) will use the VIPA address as its local address. This provides for maximum resilience against the failure of any one physical interface. SOURCEVIPA is mandatory for Enterprise Extender connections. 1

- IBMTR is the OSA link that provides the physical connectivity to the TCP/IP network for our link 2
- The VIPA address needs a virtual device and link defined; we have named it VIPA39A. The VIPA address looks like an interface to a virtual network that is concealed behind this MVS TCP/IP acting as a router. 3
- The Enterprise Extender connection to VTAM also requires a device and link definition. It uses the same interface as to another TCP/IP stack on the same MVS image, known as Samehost. TCP/IP recognizes the name IUTSAMEH as being a Samehost interface, and treats it in a fashion similar to an MPC connection (hence the device type MPCPTP). TCP/IP uses VTAM's common DLC connection manager for communication across MPC links; in the case of the Samehost interface, VTAM will define a TRLE dynamically for use by TCP/IP. 4
- We assign the address 172.16.232.39 to the VIPA interface. This is the target address configured in the Cisco SNASw router as shown in Figure 73 on page 125. 5
- The address 172.16.223.40 is assigned to the internal Enterprise Extender interface. 6

Chapter 5. Data Link Switching

This chapter shows how additional Cisco routers can act as Data Link Switching (DLSw) peers in an DLSw infrastructure made up of IBM routers.

We have to choose a version of the DLSw standard that is supported on both IBM and Cisco routers. Clearly, both implement DLSw V1 as specified in RFC 1795[1]. However, DLSw V1 has a considerable scalability problem, since TCP connections between all DLSw peers have to be established, resulting in n*(n-1)/2 bidirectional TCP connections or even n*(n-1) TCP connections when unidirectional TCP connections are used. This full mesh is necessary to transport broadcast traffic created by MAC address or NetBIOS name resolution on the underlying unicast network. This results both in considerable resource consumption on the routers for the multitude of TCP connections and also in duplication of broadcast traffic in the WAN.

RFC 2166[2] (DLSw V2) addresses the drawbacks of DLSw V1 by making use of the multicast capabilities of the underlying IP network that is used to transport the DLSw traffic. Cisco DLSw+, created in 1995, overcame the limitations of the DLSw V1 standard and provided additional functions to increase the overall scalability of DLSw. The most significant optimization feature in DLSw+ is a feature known as *peer groups*. Peer groups address broadcast replication problems that can occur in a fully meshed DLSw networks. With DLSw+, a cluster of routers can be combined into a peer group. Within a peer group, one or more of the routers is designated to be the border peer. Instead of all routers acting as peers to each another, each router within a group is a peer to the border peer and border peers establish peer connections with each other. When a DLSw+ router receives an explorer frame, it sends a single explorer frame to its border peer. The border peer checks its local, remote, and group cache for any reachability information before forwarding the explorer. If no match is found, the border peer forwards the explorer on behalf of the peer group member. If a match is found, the border peer sends the directed explorer to the appropriate peer or border peer. This setup eliminates duplicate explorers on the access links and minimizes the processing required in access routers.

RFC 2166 (DLSw V2) also addresses the drawbacks of DLSw V1 by making use of the multicast capabilities of the underlying IP network that is used to transport the DLSw traffic. Both Cisco IOS (Cisco IOS Release 11.3 and higher) and IBM implement DLSw V2, which allows Cisco routers to interoperate with IBM routers and overcome the limitations of the DLSw V1

[1] Data Link Switching: Switch-to-Switch Protocol, April 1995

[2] DLSw V2.0 Enhancements, June 1997

standard. DLSw V2 provides a scalable migration path in a mixed IBM/Cisco DLSw environment. A comparison of features offered by DLSw+ and the corresponding DLSw V2 features can be found in Appendix A, “DLSw+ And DLSw V2 features compared” on page 155.

In a mixed environment of IBM and Cisco DLSw routers we should be able to profit from all the advantages of DLSw V2, which consist of:

- Extensive use of the multicast capabilities of the underlying transport network to discover peers (address resolution) and to transmit broadcast frames (for example, NetBIOS broadcasts)
- TCP connections on demand, that is, they are established when needed and brought down after a defined period of inactivity
- Mandatory use of bidirectional TCP connections
- No need to preconfigure peers, because only a multicast address has to be configured

For our interoperability tests, first we have to set up an IP multicast-capable network - see 5.1, “Setting up IP multicast” on page 134. And secondly, we have to configure DLSw V2 on both the Cisco and IBM routers - see 5.2, “Configuring DLSw V2” on page 138.

5.1 Setting up IP multicast

Figure 80 on page 136 shows the transport network for our DLSw test. In order to make DLSw work, this IP network has to transport IP multicast packets, that is, IP packets with addresses in the range of 224.0.0.0 to 239.255.255.255. To implement IP multicast in a network, we need – compared to a “unicast” IP network – two additional protocols:

1. A group membership protocol that enables IP hosts to tell multicast-capable routers that they want to participate in a certain multicast group and thus the router should forward the IP packets destined for this group. Note that any host can *send* multicast packets, the requirement to join a multicast group is if the host wants to *receive* them.

   We will use the Internet Group Membership Protocol (IGMP) in our test scenario, since it is implemented on both IBM and Cisco routers.

2. A multicast routing protocol that tells multicast routers where to forward IP multicast packets. There is a wide range of multicast protocols available:

   a. Distance Vector Multicast Routing Protocol (DVMRP), which is the multicast routing protocol used in the MBONE of the Internet. DVMRP is supported on IBM routers; Cisco's implementation of PIM (see below) can interact with DVMRP routers.

b. Multicast Extensions to OSPF (MOSPF) is available on IBM routers but not on Cisco routers. MOSPF employs a unicast routing protocol that requires each router in a network to be aware of all available links. An MOSPF router calculates routes from the source to all possible group members for a particular multicast group. MOSPF routers include multicast information in OSPF link states. MOSPF calculates the routes for each source/multicast group pair when the router receives traffic for that pair, and routes are cached until a topology change occurs. MOSPF then recalculates the topology. Several MOSPF implementation issues have been identified and require consideration. First, MOSPF works only in networks that use OSPF. In addition, MOSPF is best suited for environments with relatively few active source/group pairs. MOSPF can take up significant router CPU bandwidth in environments that have many active source/group pairs or that are unstable.

c. Protocol Independent Multicast (PIM). PIM comes in two flavors, as PIM DM (dense mode) and PIM SM (sparse mode). Cisco routers support both of them; IBM routers support PIM DM (M[A,R]S 3.4 or later). PIM DM (which IBM and Cisco both support) initially floods all branches of the network with data, then prunes branches with no multicast group members. PIM-DM is most effective in environments where it is likely that there will be a group member on each subnet. PIM-DM assumes that the multicast group members are densely distributed throughout the network and that bandwidth is plentiful. PIM SM (which Cisco supports) can be used for any combination of sources and receivers, whether densely or sparsely populated, including topologies where senders and receivers are separated by WAN links, and/or when the stream of multicast traffic is intermittent.

The only multicast routing protocol currently supported on both Cisco and IBM routers is PIM DM. However, PIM DM is not tailored for use in a WAN. A PIM DM router forwards multicast packets on its interfaces by default until a neighbor router tells it that there are no members of a specific multicast group on that interface. PIM SM behaves the other way around: neighbor routers have to indicate that they are interested in certain multicast groups before their multicast traffic is forwarded to them. Since most DLSw networks use a WAN as part of their transport network, which makes bandwidth a scarce resource, we are not going to choose PIM DM as the multicast protocol. Instead, we will use DVMRP in the IBM router backbone and attach the additional Cisco routers with the DVMRP-PIM interworking function.

DVMRP offers the possibility of establishing tunnels through IP backbones that do not support IP multicasting. This property might be useful if you are adding Cisco backbone routers to your IP network that are not DLSw peers and thus not endpoints of multicast traffic.

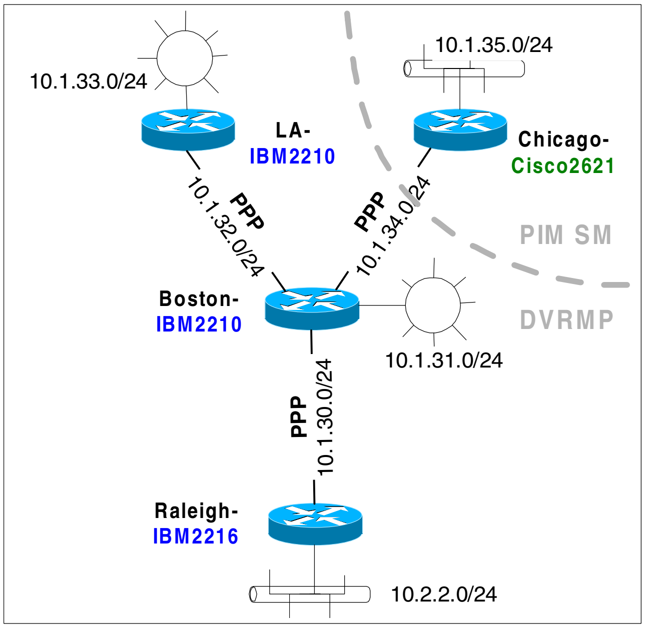


Figure 80. Multicast IP network

In Figure 81 on page 137 we show how to set up multicasting and DLSw V2 on Chicago-Cisco2621:

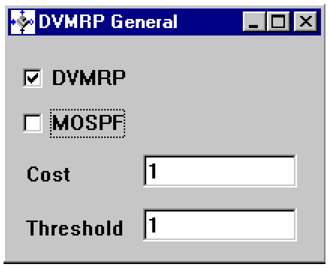
- First we turn on IP multicast routing **1**.
- Then we enable the serial interface to receive IP multicast topology updates **6** from the DVMRP area. There is no special command to configure interworking between DVMRP and PIM SM; PIM SM routers automatically listen to DVMRP topology updates.
- In order to receive multicast traffic destined for the multicast group 224.0.10.0, which is the multicast group the IETF recommends for DLSw V2, at least one interface on the router has to join the group **7**.

To configure DVMRP on the IBM routers LA-IBM2210, Boston-IBM2210, and Raleigh-IBM2216, enable DVMRP using the Protocols/IP/DVMRP/General menu of the configuration tool as shown in Figure 82 on page 137. Then enable DVMRP on the serial interfaces that interconnect the routers acting as

DLSw peers in the Protocols/IP/DVMRP/Interfaces menu as shown in Figure 83 on page 138. There is no need to enable DVMRP on the LAN interfaces.

```
ip multicast-routing 1
!
dlsw local-peer peer-id 10.1.255.32 promiscuous 2
dlsw bridge-group 1 3
dlsw multicast 224.0.10.0 4
!
interface Loopback0
 ip address 10.1.255.32 255.255.255.255
!
interface FastEthernet0/0
 ip address 10.1.35.32 255.255.255.0
 bridge-group 1 5
!
interface Serial0/0
 mtu 2048
 encapsulation ppp
 ip address 10.1.34.32 255.255.255.0
 ip pim sparse-mode 6
 ip igmp join-group 224.0.10.0 7
!
router rip
 version 2
 network 10.0.0.0
!
bridge 1 protocol ieee 8
```

Figure 81. DLSw V2 configuration for a Cisco router



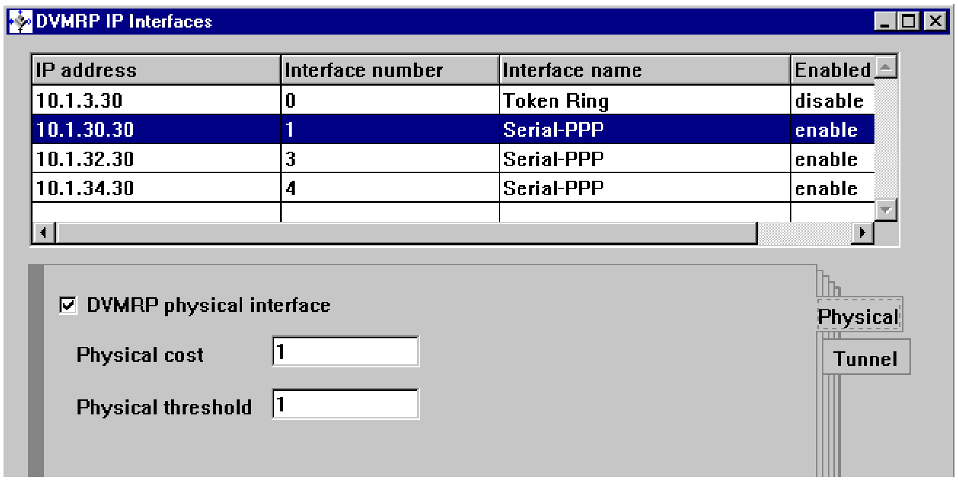
☑ DVMRP

□ MOSPF

Cost: 1

Threshold: 1

Figure 82. Enabling DVMRP on IBM routers



DVMRP IP Interfaces

IP address	Interface number	Interface name	Enabled
10.1.3.30	0	Token Ring	disable
10.1.30.30	1	Serial-PPP	enable
10.1.32.30	3	Serial-PPP	enable
10.1.34.30	4	Serial-PPP	enable

☑ DVMRP physical interface

Physical cost: 1

Physical threshold: 1

Figure 83. Enabling DVMRP on the Interfaces of IBM routers

5.2 Configuring DLSw V2

In order to test the DLSw V2 interoperability of IBM and Cisco routers we set up the network shown in Figure 84 on page 139. The Windows workstations on the LANs of IBM-LA2210 and Chicago-Cisco2621 communicate using NetBIOS, and the two APPN Network Nodes establish a CP-CP session using LLC2.

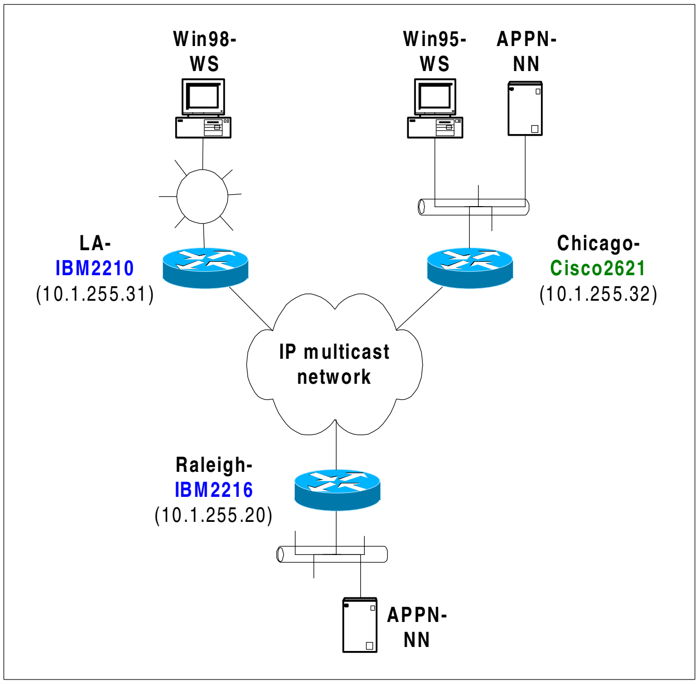
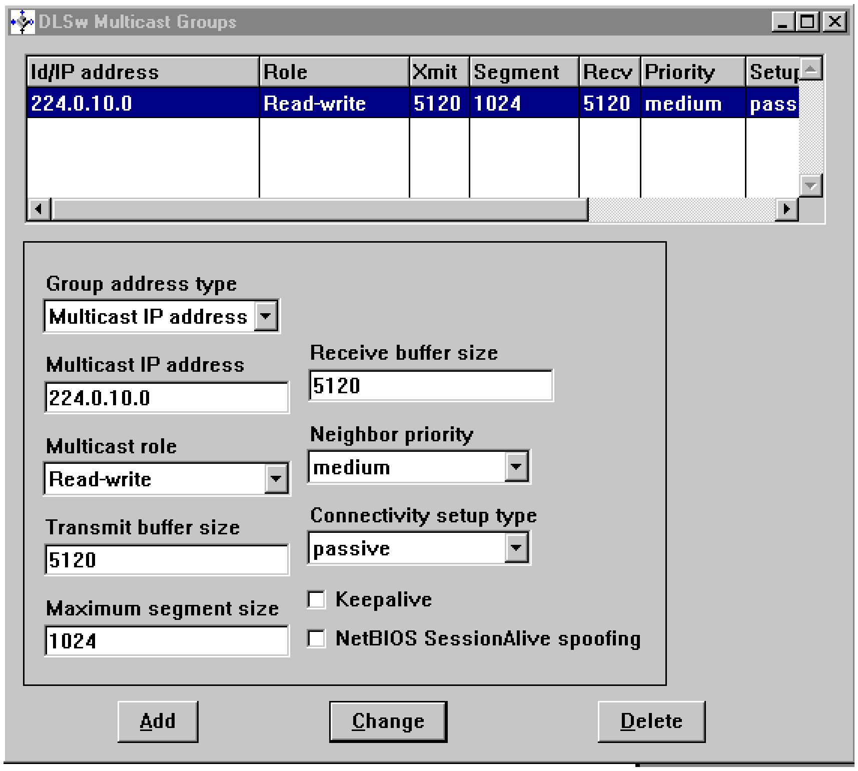


Figure 84. DLSw test network

The configuration of DLSw V2 on the IBM routers is as follows:

- Bridging is enabled on the LAN ports whose NetBIOS/SNA traffic has to be transported by means of DLSw. On the token-ring interface of LA-IBM2210 source route bridging has to be enabled; on the Ethernet interface of Raleigh-IBM2216 transparent bridging has to be turned on.
- DLSw uses the *internal* IP addresses of the IBM routers, which are shown in Figure 84.
- DLSw has to be enabled, a DLSw token-ring segment has to be defined, and the SAPs for NetBIOS and SNA have to be opened on the LAN interfaces. There is no need to define any neighbors manually. Instead we define a multicast group for DLSw in the Protocols/IP/DLSw/Multicast Groups menu as shown in Figure 85 on page 140. The group address type must be Multicast IP address, which allows the explicit definition of the IP multicast address used for DLSw. The definition by means of group IDs is IBM proprietary and will not work together with the Cisco router. In this example we chose 224.0.10.0 as multicast address for DLSw, since this is the address recommended in RFC 2166.

The connectivity setup type is “passive”, which means that TCP connections between peers will be only established when data has to be transferred. This implies that dynamic neighbors have to be admitted in order to accept the establishment of TCP connections from foreign DLSw peers.



DLSw Multicast Groups

Id/IP address	Role	Xmit	Segment	Recv	Priority	Setu
224.0.10.0	Read-write	5120	1024	5120	medium	pass

Group address type: Multicast IP address

Multicast IP address: 224.0.10.0

Multicast role: Read-write

Transmit buffer size: 5120

Maximum segment size: 1024

Receive buffer size: 5120

Neighbor priority: medium

Connectivity setup type: passive

□ Keepalive

□ NetBIOS SessionAlive spoofing

Figure 85. Definition of DLSw multicast group

The DLSw V2 configuration of Chicago-Cisco2621 is contained in Figure 81 on page 137:

- DLSw is started on a Cisco router by means of the `dlsw local-peer` command, where we define the IP address of the local DLSw peer. The keyword `promiscuous` corresponds to the dynamic neighbors feature on IBM routers and allows the router to accept incoming TCP connections. 2
- The Ethernet interfaces on the Cisco router have to be assigned to a *bridge group*. A bridge group contains all interfaces between which frames are bridged by a specific bridge process. With `dlsw bridge-group 1` we link DLSw to bridge group 1. 3

- We configure the IP multicast address to be 224.0.10.0 as on the IBM routers 4.
- We link the Fast Ethernet interface to bridge group 1, which DLSw is already linked to. 5
- The spanning tree protocol used in bridge group 1 is IEEE 802.1. 8

The network configured as described above successfully transported NetBIOS and SNA traffic between both Windows desktops and both APPN networks. However, we once encountered an error that we were not able to reproduce later: when performing a repeated APPN ping between the APPN nodes, all operations on the router Chicago-Cisco2621 stopped for a few seconds every 10 seconds. Moreover, on the console of the Cisco router, the error message `%DLSWP-3-PNOCOOKIE: DLSw: uninitialized peer read from 10.1.255.31(1031) to 10.2.255.32(2067)` appears after a TCP connection with an IBM peer is established, which does not seem to affect interoperability between IBM and Cisco routers. The `%DLSWP-3-PNOCOOKIE` messages seems to be a side effect of another Cisco bug (CSCdm30793) which was fixed in Cisco IOS 11.3 and 12.0. It has been brought to the attention of Cisco DLSw+ development for further investigation and resolution.

5.3 Assuring QoS for DLSw traffic

Now that we have successfully set up a multicast capable IP network for the transport of DLSw V2, we want to make sure that the DLSw traffic receives at least 50% of the available bandwidth on the WAN links throughout the network. Our approach to prioritize SNA traffic makes use of the precedence bits in the TOS field of the IP packet header, thus allowing interoperability with DiffServ-aware transport networks. Our approach requires two operations:

- DLSw peers have to mark the IP packets they are using to communicate. In this example we will use the precedence value 011 (“flash” according to RFC 791[3]) to mark the IP packets using TCP source or destination port 2065 or 2067 and UDP source or destination port 2067. These TCP/UDP ports are used by DLSw V2.
- The WAN links of the DLSw peer routers and of the routers in the transport network must recognize IP packets whose precedence is 011 and assure that these packets get at least 50% of the bandwidth of the WAN links.

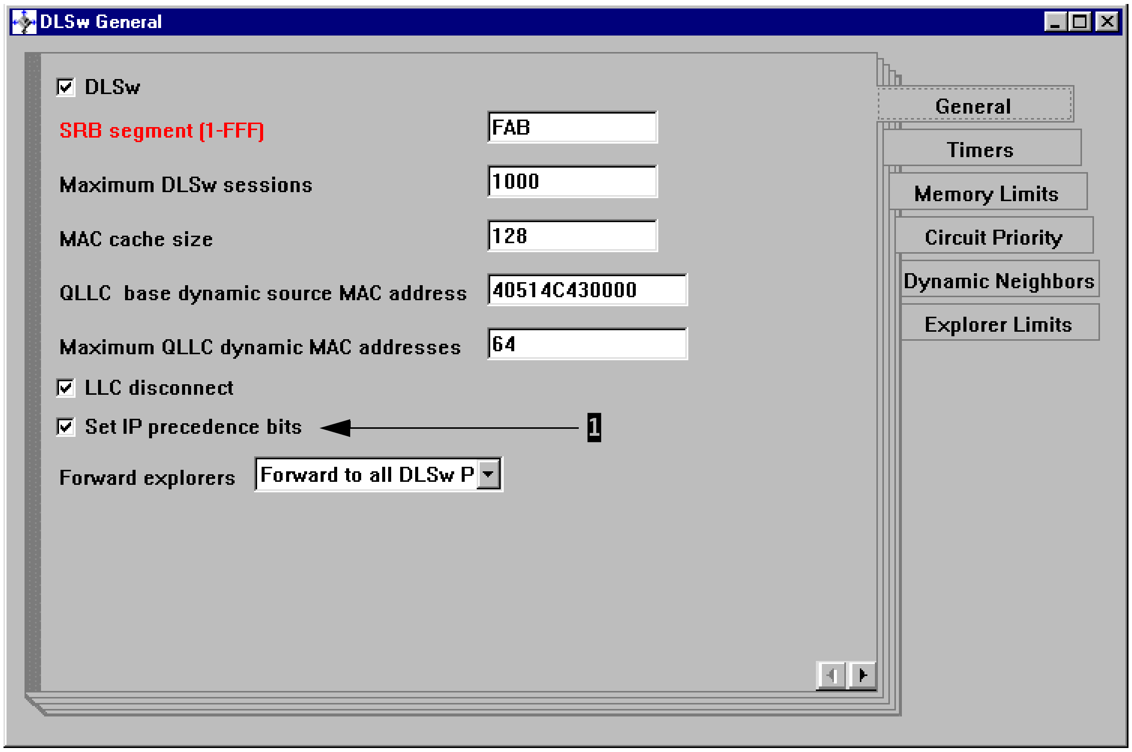
In 5.3.1, “QoS for DLSw on an IBM router” on page 142 will show how this is done for an IBM DLSw peer. In 5.3.2, “QoS for DLSw on a Cisco router” on

[3] But see page 56 for a discussion of the different interpretation of “precedence” by IBM and Cisco

page 144, we will show what a similar configuration looks like on a Cisco router.

5.3.1 QoS for DLSw on an IBM router

First we have to make the IBM router set the precedence bits to 011, which is shown in Figure 86. 1 In the Protocols/DLSw/General panel we just click the **Set IP Precedence Bits** check box. Note that there is no control to which precedence value DLSw traffic is set. The IBM router will always set the first three bits of the TOS field for DLSw traffic to “011”. If we wanted to set other precedence values, we would have to filter the IP packets of the DLSw traffic by means of access controls and explicitly set their precedence bits.



☑ DLSw

SRB segment (1-FFF) FAB

Maximum DLSw sessions 1000

MAC cache size 128

QLLC base dynamic source MAC address 40514C430000

Maximum QLLC dynamic MAC addresses 64

☑ LLC disconnect

☑ Set IP precedence bits

Forward explorers Forward to all DLSw P

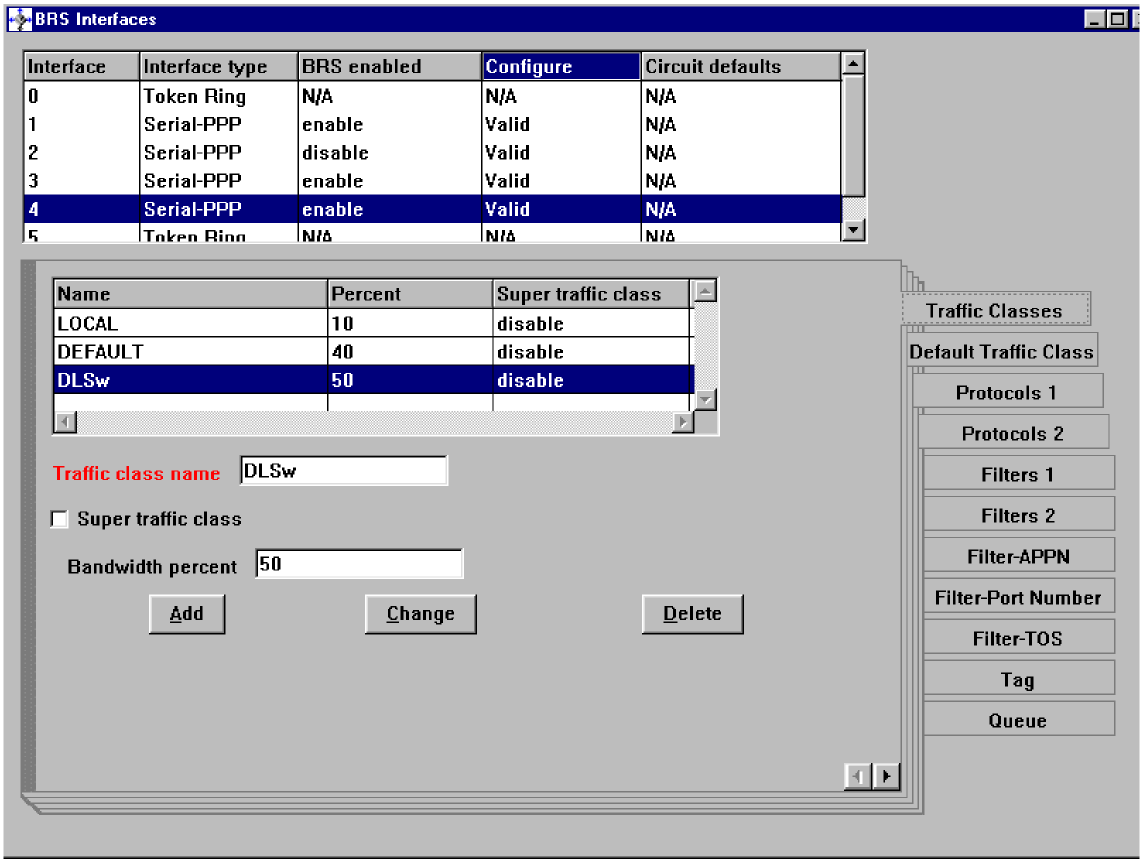
Figure 86. Setting precedence bits for DLSw traffic on IBM routers.

Next we configure BRS on the WAN links to ensure that traffic with IP precedence “011” gets at least 50% of the bandwidth. We define a traffic class DLSw, as shown in Figure 88 on page 143. Then we use the TOS filters to select the IP packets with precedence 011 and assign them to traffic class DLSw, as shown in Figure 89 on page 144. To understand the TOS mask and the TOS range setting, you need to be aware of the structure of the TOS byte according to RFC 1349

```
   0     1     2     3     4     5     6     7
+-----+-----+-----+-----+-----+-----+-----+-----+
|                 |                       |     |
|   PRECEDENCE    |          TOS          | MBZ |
|                 |                       |     |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

Figure 87. Structure of the Service Type byte in the IP header according to RFC 1349

The three most significant bits carry the precedence value, which is only part of the Service Type byte we are interested in. We pick those bits with the TOS mask 0xE0 = 1110 0000 and check whether the result is 0110 0000 = 0x60 by setting the TOS range from 0x60 to 0x60.



BRS Interfaces

Interface	Interface type	BRS enabled	Configure	Circuit defaults
0	Token Ring	N/A	N/A	N/A
1	Serial-PPP	enable	Valid	N/A
2	Serial-PPP	disable	Valid	N/A
3	Serial-PPP	enable	Valid	N/A
4	Serial-PPP	enable	Valid	N/A
5	Token Ring	N/A	N/A	N/A

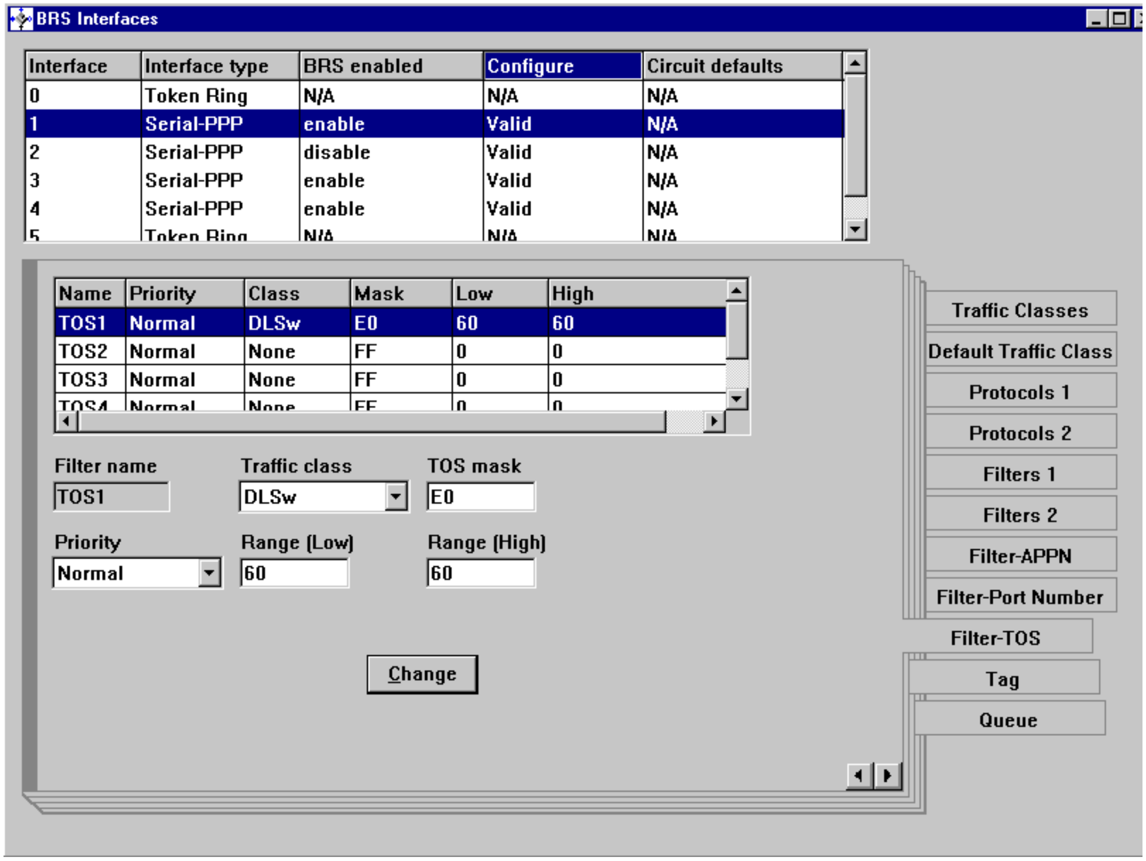
Name	Percent	Super traffic class
LOCAL	10	disable
DEFAULT	40	disable
DLSw	50	disable

Traffic class name: DLSw

☐ Super traffic class

Bandwidth percent: 50

Figure 88. Definition of a traffic class for DLSw traffic



Interface	Interface type	BRS enabled	Configure	Circuit defaults
0	Token Ring	N/A	N/A	N/A
1	Serial-PPP	enable	Valid	N/A
2	Serial-PPP	disable	Valid	N/A
3	Serial-PPP	enable	Valid	N/A
4	Serial-PPP	enable	Valid	N/A
5	Token Ring	N/A	N/A	N/A

Name	Priority	Class	Mask	Low	High
TOS1	Normal	DLSw	E0	60	60
TOS2	Normal	None	FF	0	0
TOS3	Normal	None	FF	0	0
TOS4	Normal	None	FF	0	0

Filter name: TOS1

Traffic class: DLSw

TOS mask: E0

Priority: Normal

Range (Low): 60

Range (High): 60

Figure 89. Assign traffic with precedence 011 to the DLSw traffic class

5.3.2 QoS for DLSw on a Cisco router

Now we try to imitate the treatment of DLSw traffic on the IBM router with a similar configuration on a Cisco router. Figure 90 on page 145 shows the additional commands you need to configure QoS on Chicago-Cisco2621.

```
dlsw tos map high 3 medium 3 normal 3 low 3 1
!
access-list 101 permit ip any any precedence flash 2
!
class-map DLSwTraffic 3
 match access-group 101 4
!
policy-map DLSwRes 5
 class DLSwTraffic 6
   bandwidth 32 7
!
interface Serial0/0
 bandwidth 64000 8
 no fair-queue 9
 service-policy output DLSwRes 10
!
```

Figure 90. QoS configuration for DLSw traffic on a Cisco router

1. The first step is to set the precedence of IP packets containing TCP and UDP information for DLSw to “011”. This is done by the command `dlsw tos map`. 1 On Cisco routers you can have up to four TCP connections between two peers which are carrying DLSw traffic of different priorities. For each of the priorities you specify a specific precedence value. In our case we set them all to 011 = 3 to imitate the behavior of the IBM routers.

2. Next, we need to find a mechanism similar to BRS that allocates a minimum bandwidth on the outbound WAN interface of the router. Cisco offers two queuing techniques for this purpose: Custom Queuing (CQ) and Class Based Weighted Fair Queuing (CBWFQ), which are both available in IOS release 12.0(5) or later. CBWFQ is easier to configure with no loss of function when compared with CQ. Thus we chose CBWFQ for our example.

3. CBWFQ applies different service policies to distinct traffic classes. The first thing to do is to define traffic classes. In our case we need just one class, which is the class of IP traffic with precedence “011”. The commands 2, 3, and 4 in Figure 90 select this traffic and put it into class DLSwTraffic.

4. The filter definition in line 2. shows the different approach of Cisco concerning the interpretation of the service byte of the IP header. Cisco assumes a structure conforming to RFC 1349 (see Figure 87 on page 143) and uses such keywords as `precedence` and `tos` to access the contents of the service byte, whereas IBM routers make no assumption about the structure of the Service Type byte.

5. Then a service policy for the DLSwTraffic class has to be defined, which is accomplished in 5, 6, and 7. The name of the service policy is DLSwRes 5.
6. We assign the DLSwTraffic class to this policy and allocate a minimum bandwidth of 32 kbps 7. Since the service policy is not attached to specific interface, we have to specify the absolute bandwidth compared to a relative specification as a percentage of the interface bandwidth 8 on the IBM routers with BRS.
7. Finally, we attach the service policy DLSwRes to the only WAN interface of Chicago-Cisco2621 10, thereby enabling CBWFQ on this interface.
8. Note that you have to turn off WFQ (9), which is the default queuing technique on interfaces with a bandwidth 2 Mbps or less. Otherwise, the router will not accept the CBWFQ definitions.

Chapter 6. IPX interoperability

To do some basic IPX interoperability testing we build the IPX network shown in Figure 92 on page 148. The IPX network consists exclusively of IPX broadcast circuits. The host numbers of the routers are indicated in brackets. On Cisco routers, the host number is, unless explicitly set, the MAC address of the first LAN interface. On the IBM routers it has to be set by means of the `SET HOST-NUMBER` command in the IPX configuration dialog. The IPX configuration of the IBM routers is straightforward, thus we just show the configurations of Chicago-Cisco2621 and Bonn-Cisco2621, which can be found in Figure 91 on page 147 and Figure 93 on page 148 respectively:

- First we enable IPX routing on the router. 1
- Then we assign IPX network numbers to the Ethernet and serial interfaces. 2, 3
- We found that inverse ARP for IPX addresses on frame relay links between IBM and Cisco routers does not work properly. Therefore we had to manually assign IPX addresses to frame relay DLCIs and we turned off inverse ARP for IPX on Bonn-Cisco2612 (4) and then assigned the IPX address of the router at other end of the PVC to DLCI 272(5). Note that you also have to do this manual IPX address mapping on the IBM router (we don’t show this here). This problem has now been fixed, see CSCdp94497 and NA06204 for fixes to Cisco and IBM code respectively.

```
ipx routing 1
!
interface FastEthernet0/0
 speed 10
 ipx network EE35 2
!
interface Serial0/0
 mtu 2048
 bandwidth 64000
 encapsulation ppp
 ipx network AA34 3
```

Figure 91. IPX configuration of Chicago-Cisco2621

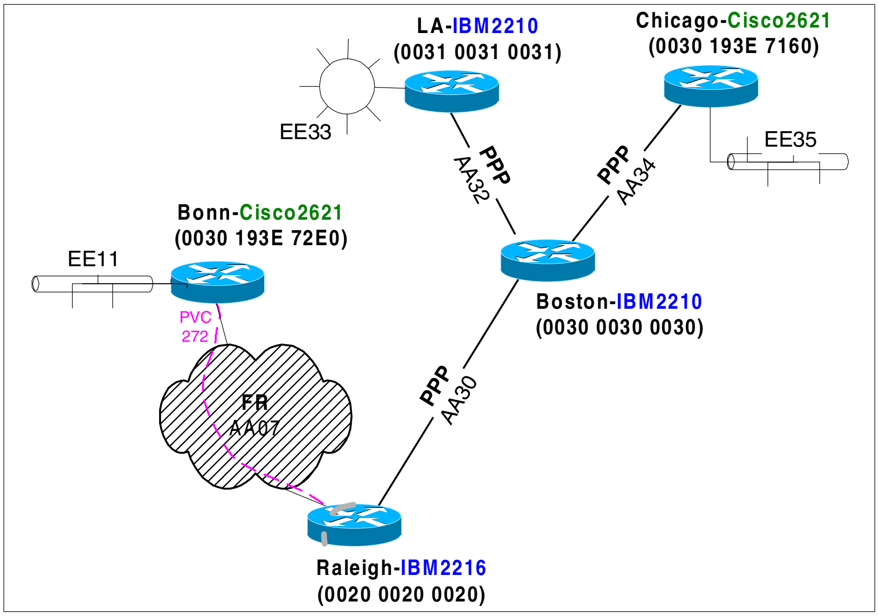


Figure 92. IPX test network

```
ipx routing 1
!
interface FastEthernet0/0
 duplex auto
 speed auto
 ipx network EE11 2
!
interface Serial0/0
 mtu 2048
 encapsulation frame-relay IETF
 frame-relay lmi-type ansi
 ipx network AA07 3
 no arp frame-relay 4
 frame-relay map ipx AA07.0020.0020.0020 272 5
```

Figure 93. IPX configuration of Bonn-Cisco2621

Figure 94 on page 149 shows the result of the inverse ARP procedure, when the Cisco router is trying to learn the IPX address on the other end of the frame relay PVC. Note that for some reason, the Cisco router thinks the IPX address at the other end of the link is `200020.0020.4421.0000` whereas the correct address is `AA07.0020.0020.0020`.

```
Bonn-Cisco2621#show frame-relay map
Serial0/0 (up): ipx 200020.0020.4421.0000 dlci 272(0x110,0x4400),dynamic,
broadcast, IETF, status defined, active
```

Figure 94. Wrong result of inverse ARP for IPX address on frame relay

With the manual assignment of the IPX addresses on frame relay PVCs the IPX network transported IPX traffic without further problems. An example of a routing table of the running IPX network (from Bonn-Cisco2621) is shown in Figure 95.

```
Bonn-Cisco#show ipx route
Codes: C - Connected primary network,    c - Connected secondary network
       S - Static, F - Floating static, L - Local (internal), W - IPXWAN
       R - RIP, E - EIGRP, N - NLSP, X - External, A - Aggregate
       s - seconds, u - uses, U - Per-user static

7 Total IPX routes. Up to 1 parallel paths and 16 hops allowed.

No default route known.

C       AA07 (FRAME-RELAY),   Se0/0.1
C       EE11 (NOVELL-ETHER),  Fa0/0
R       AA30 [04/01] via     AA07.0020.0020.0020,   57s, Se0/0.1
R       AA32 [04/02] via     AA07.0020.0020.0020,   57s, Se0/0.1
R       AA34 [04/02] via     AA07.0020.0020.0020,   57s, Se0/0.1
R       EE33 [05/03] via     AA07.0020.0020.0020,   57s, Se0/0.1
R       EE35 [10/03] via     AA07.0020.0020.0020,   57s, Se0/0.1
```

Figure 95. IPX routing table of Bonn-Cisco2621

We encountered another IPX interoperability problem when trying to ping from Cisco to IBM routers. It turned out that when we try to ping an IBM router from a Cisco router running IOS 12.0(4), the IBM router does not respond to that IPX ping. For example, see Figure 96 on page 150 where we ping from Chicago-Cisco2621 running IOS 12.0(4) to Boston-IBM2210 without success 2. Note that the Cisco router is sending “IPXcisco Echos”. Pinging from Bonn-Cisco2621 which is running 12.0(7) to Boston-IBM2210 works without problems 1. Note that the Cisco router is now sending “IPX Novell Echoes”.

```
Bonn-Cisco#ping ipx
Target IPX address: aa34.0030.0030.0030
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]:
Type escape sequence to abort.
Sending 5, 100-byte IPX Novell Echoes to AA34.0030.0030.0030, timeout is 2 seconds: 1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 100/118/128 ms
Bonn-Cisco#telnet 10.1.255.32
Trying 10.1.255.32 ... Open


User Access Verification

Password:
Chicago-Cisco2621>en
Password:
Chicago-Cisco2621#ping ipx
Target IPX address: aa34.0030.0030.0030
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Verbose [n]:
Type escape sequence to abort.
Sending 5, 100-byte IPXcisco Echoes to AA34.0030.0030.0030, timeout is 2 seconds 2:
.....
Success rate is 0 percent (0/5)
```

Figure 96. Different versions of IPX PING on Cisco routers

Chapter 7. Dial-in connectivity

This chapter provides a very brief overview of the use of a Cisco router as a dial-in server, especially for those people accustomed to the similar set of functions available on the IBM router family.

Figure 97 shows our dial-in configuration:

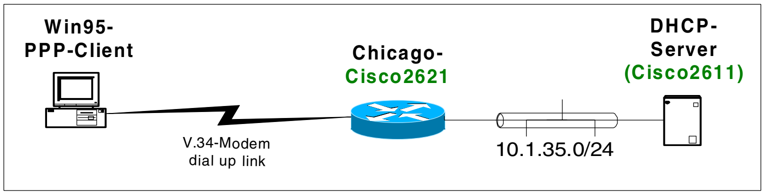


Figure 97. Dial-up network environment

The configuration of the Cisco router as a dial-in server is shown in Figure 99 on page 153:

- The local username/password database is used for authentication of incoming PPP connections. 1
- The `username` command populates the local username/password database.
- The `aux` port of the Cisco router is configured for the attachment of a V.34 modem 7 and told to expect PPP connections on it 8. The other physical properties of the `aux` port, such as hardware handshaking and speed, are configured below.
- The internal line number of the aux port on this Cisco router is 65, which you can find out by means of the `show line` command. Note that the line numbering is different on each router model. On top of the physical line 65 we define the (logical) asynchronous interface 65 4. Incoming PPP connections are authenticated using the MS-CHAP protocol 6. The IP address assigned to the PPP peer is retrieved from a DHCP server 5.
- The DHCP server to contact when asynchronous interfaces ask for peer IP addresses is 10.1.35.50. 3.

The configuration of the Cisco router as DHCP server is shown in Figure 98 on page 152:

- A pool of IP addresses to be handled by the DHCP server is defined in 2 and 3. The associated DHCP options that are passed to the DHCP clients are defined in 4.

© Copyright IBM Corp. 2000

- No persistent external database is used to keep track of the DHCP leases. 1

```
no ip dhcp conflict logging 1
ip dhcp excluded-address 10.1.35.1 10.1.35.250 2
!
ip dhcp pool Ether35 3
   network 10.1.35.0 255.255.255.0
   default-router 10.1.35.32 4
   dns-server 77.77.77.77 88.88.88.88 99.99.99.99
   netbios-name-server 3.3.3.3
   netbios-node-type m-node
!
interface Ethernet0/0
 ip address 10.1.35.50 255.255.255.0
 no ip directed-broadcast
```

Figure 98. Configuration of the DHCP server

```
aaa new-model
aaa authentication login RouterConsole none
aaa authentication ppp default local 1
enable secret 5 $1$dTDw$O9c9ji7548XPl/hSZnnaA.
!
username mcschmid password 0 geheim 2
username Chicago-Cisco2621 password 0 geheim
!
ip dhcp-server 10.1.35.503
!
interface Loopback0
 ip address 10.1.255.32 255.255.255.255
!
interface FastEthernet0/0
 ip address 10.1.35.32 255.255.255.0
 no ip directed-broadcast
 duplex auto
 speed 10
!
interface Serial0/0
 mtu 2048
 bandwidth 64000
 ip address 172.16.1.2 255.255.0.0
 no ip directed-broadcast
 encapsulation ppp
 no ip mroute-cache
 fair-queue 64 256 0
!
interface Async65 4
 ip unnumbered FastEthernet0/0
 no ip directed-broadcast
 encapsulation ppp
 async mode interactive
 peer default ip address dhcp 5
 fair-queue 64 16 0
 ppp authentication ms-chap 6
!
router rip
 version 2
 network 10.0.0.0
 network 172.16.0.0
 no auto-summary
!
line aux 0 7
 autoselect ppp 8
 modem InOut
 transport input all
 speed 115200
 flowcontrol hardware
```

Figure 99. Configuration of the dial-in server

Using an IBM2210 as DHCP server

We have imitated the configuration of the Cisco DHCP server on an IBM2210. However, the IBM encounters problems when processing the DHCP Discover Request from the Cisco router, thus failing to communicate an IP address (and other options) to the dial-in server for the PPP client. The DHCP debugging on the IBM DHCP server shows the error message:

```
DHCP.234: Failed to allocate buffer for reply message, size 1124.
```

Appendix A. DLSw+ And DLSw V2 features compared

This appendix compares Cisco DLSw+ features and implementation with IBM's DLSw features and implementation using the DLSw V2 (RFC 2166) standard.

DLSw V2 (RFC 2166) is supported by both IBM and Cisco, and is the preferred DLSw platform for interoperability between IBM and Cisco hardware platforms.

Dynamic Peers

Dynamic peers are configured remote peers that are only connected when required. Cisco supports both unicast and multicast methods of finding the dynamic peers. Cisco supported dynamic peers (also called on-demand peers) in the first release of DLSw+. Cisco submitted this capability to the DLSw Related Interest Group of the APPN Implementers Workshop, and it was added to the DLSw standard in V2 RFC 2166.

Cisco's initial DLSw+ implementation, however, was based on a border peer implementation, and the means to find the dynamic peers was different (Cisco DLSw+ used unicast, while the V2 standard specified multicast). However, Cisco implements full DLSw V2 compliant capabilities (which includes sending explorers via unicast and multicast UDP).

The IBM 221x DLSw V2 equivalent of this function is the ability to configure a DLSw partner, partners within a group, or dynamic partners as passive. IBM supports sending of explorers via unicast and multicast UDP.

Border Peers

Border peers is a Cisco two-level DLSw peer architecture designed to solve DLSw scalability problems inherent in the DLSw V1 (RFC 1795) standard. Cisco proposed this architecture to the DLSw Related Interest Group of the APPN Implementers Workshop as a possible standard solution to these scalability problems. The RIG chose to use multicast IP instead of border peers, and the multicast IP scalability solution was standardized in DLSw Version 2, RFC 2166 (which Cisco also supports).

IBM 221x DLSw uses the DLSw V2 standard (multicast IP) and supports it fully. IBM does not support Cisco's DLSw+ border peer implementation.

On-Demand Peers

On-demand peers are peer connections that are established without pre-configuration. Cisco DLSw+ implements the on-demand peers capability using the DLSw+ border peer function. Once a Cisco border peer learns about a resource, it will directly contact that resource using the on-demand peer that was established.

The DLSw V2 equivalent function (which IBM and Cisco both support) is via the use of multicast DLSw groups to discover DLSw peers within the same multicast DLSw group. This function is a simple extension of the DLSw Version 2 standard.

LNM, DSPU, or NSP over DLSw

IBM's LAN Network Manager is a management tool used to manage token-ring media attachment units (MAUs) and token-ring adapters. It uses a proprietary protocol to communicate with agent software in source-route bridges and in Cisco routers to obtain the status of the token-ring network and to send commands to token-ring-attached devices.

There are no special configuration requirements to use LAN Network Manager in conjunction with Cisco DLSw+. Cisco DLSw+ supports any LNM capabilities that do not require a full RIF. In the few instances where the full RIF is required, Cisco has provided a DLSw+ feature extension (RIF passthru) that does not terminate the RIF.

IBM 221x DLSw (V2) is also able to transport LNM traffic via DLSw. The target LNM may also be in the same box as DLSw. The IBM 221x DLSw support also includes transport of LNM-to-LSM and LNM-to-hub traffic.

Cisco also has the ability to transport downstream PU (DSPU), and native service point (NSP) traffic via DLSw+ (IBM does not support DSPU or NSP transport as these are Cisco-specific, IOS-related functions).

APPN over DLSw

Cisco has the ability to transport APPN traffic via DLSw+. The target APPN station may be in the same router as DLSw+ or separate. It has the capability to automatically map the APPN COS (transmission priority) to IP TOS. Cisco has several features that take advantage of the IP TOS field using advanced quality-of-service (QOS) algorithms such as RSVP, weighted fair queuing, weighted random detect, and several other techniques.

IBM 221x DLSw is also able to transport APPN via DLSw. IBM 221x DLSw also supports an internal interface to APPN in the same router. All SNA end stations in the DLSw network are visible to APPN. Both IBM and Cisco support the internal interface between APPN and DLSw by the same sort of mechanism - they both define a virtual MAC address internal to the router, and then use this MAC address as the source MAC address for frames to be sent over DLSw. So as far as the DLSw function is concerned there is no difference between APPN traffic originating from an internal APPN function of the router itself or from a LAN-attached APPN device.

When a LAN APPN device wants to connect to another APPN device, it typically establishes an LLC2 connection with the partner APPN node using TEST frames followed by SABME, in exactly the same way as LAN-attached PU2 devices establish an LLC2 connection. DLSw is perfectly capable of handling this connection just like any other SNA LLC2 connection, and will establish a circuit based on the unique MAC/SAP pairs used for the connection when the TEST frame flows. The typical and default APPN SAP used for this connection is x’04’.

If the same APPN devices are also capable of running HPR, they will often use a different SAP - x’C8’ by default - for this connection and use connectionless (LLC1) unnumbered information (UI) frames for this traffic. This is where DLSw has a problem: there is no circuit established for this new MAC/SAP pair and therefore DLSw does not know what to do with these frames. The HPR nodes themselves know the MAC addresses of their partners; indeed they are required to use the same routing information field (RIF) in token-ring networks that they already discovered for the base APPN connection, but DLSw’s MAC address cache times out (according to Cisco documentation) and therefore DLSw routers effectively “forget” how to handle these HPR frames.

If the HPR SAP and the base APPN SAP are the same, this problem does not arise. Some implementations allow this, but some do not, and in any case most implementations default to using different SAP values anyway. The HPR architecture allows the SAP values to be the same or to be different. There are no architectural rules being broken in either case, but the reason for choosing a different SAP is performance-related: the device driver in a receiving APPN station does not need to examine frames received over different SAPs before passing them up the APPN stack. If the same SAP is used then the device driver must examine each frame, essentially so that UI HPR frames are passed upwards with no need to perform link-level error recovery checking. The only case in which the SAPs are *required* to be the same is where HPR is using link-level error recover procedures (LLERP). This is not usually the case because HPR’s ability *not* to use LLERP is one of its main benefits.

Cisco’s DLSw+ implementation (since IOS 11.3) provides for the support of LLC1 UI frames in that a new DLSw circuit will be established for a specifically routed UI frame for a destination MAC address about which DLSw already “knows”. These are referred to as “lightweight” circuits because there is no active LLC2 connection (with regular “keepalive” flows) which supports the circuit; the status of the circuit will remain in CIRCUIT_ESTABLISHED state (and not in CONNECT state) until there is no UI frame flow for the MAC/SAP pair for 10 minutes.

It is not clear how these lightweight circuits are re-established after a time-out if the destination MAC address is not “known” to the Cisco DLSw+ router. APPN/HPR links can be established without the requirement for an LLC2 connection between nodes, which can be the case both when CP-CP sessions are supported over RTP (option set 1402) or for an APPN TG simply used for the transport of data but without CP-CP sessions. Even in these cases, though, a subset of the LLC2 implementation called Logical Data Link Control (LDLC) is required and this involves the transmission of frames such as TEST and XID and associated acknowledgments. As long as the Cisco DLSw+ implementation is capable of saving the MAC address information returned for successful HPR LDLC connections, it should be possible to transport HPR reliably and successfully using DLSw+, which is almost certainly not the case using DLSw V1 or DLSw V2.

Given that both IBM and Cisco now support Enterprise Extender, this discussion is reasonably academic since Enterprise Extender should now be the preferred method of transporting APPN traffic across IP networks.

Payload Compression

Cisco DLSw+ and IBM DLSw implementations both have the ability to perform DLSw-specific payload compression. Cisco DLSw+ also support FRF.9 compression over frame relay and has hardware-assisted compression features to run compression at very high rates in the central site.

MIB Support

Both Cisco DLSw+ and DLSw V2 support the IETF standard DLSw MIB, RFC 2024.

SNA View PU Correlation

Cisco DLSw implementation provides hooks for its Cisco SNA View PU correlation platform. SNA View allows operation of all PUs and LUs in an IBM environment from an SNMP console. The DLSw+ hook enables SNA View to show which DLSw+ routers connect a given SNA PU pair and exactly which IP devices are in between.

IBM routers do not support the Cisco SNA View platform.

Multidrop 2.1

Both IBM and Cisco DLSw support multidrop PU 2.0 and PU 2.1 configurations, both as primary and secondary link stations. This includes the intermixing of PU2.0 and T2.1 devices, and also the support of group poll as a secondary link station.

80D5 Encapsulation

IBM products support the DLSw standard TCP encapsulation method, as well as the standard RFC 1490 bridged, routed, and HPR routed formats over frame relay. Frame relay BAN (bridged format) is a powerful method for multiplexing remote end station connections onto a single frame relay circuit without the overhead of DLSw TCP headers.

Cisco DLSw+ supports 80D5 IBM LAN encapsulation, as well all IEEE Ethernet LAN encapsulation methods. Cisco also supports DLSw RFC 1490 encapsulation, including BAN, BNN, and DLSw Lite (which uses DLSw state machines but only RFC1490 headers). In addition to DLSw TCP and RFC 1490 encapsulation, Cisco DLSw+ also supports Fast Sequenced Transport (FST), an IP/UDP encapsulation technique that allows very high-speed transport by using fast switching (instead of process switching) in Cisco router platforms.

Port Lists

Cisco has the ability to designate a group of LAN ports as ports supporting DLSw+, excluding other ports. Using port lists, you can control where broadcasts are forwarded by setting up defined broadcast domains. Cisco DLSw+ supports SAP, MAC address, LU and user-defined data filters.

IBM 221x DLSw equivalent of this function is the opening of DLSw SAPs on interfaces. This not only specifies whether a port supports DLSw, but which DLSw SAPs are supported on that port.

Load balancing and Fault Tolerance

Load balancing between multiple active DLSw peers on the same target LAN is the ability to bring up DLSw sessions over all active DLSw peers simultaneously to share the traffic load.

Fault tolerance between multiple active DLSw peers on the same target LAN is the ability to use secondary (capable) DLSw peer connections only if the primary (preferred) DLSw peer connection becomes unavailable.

Cisco DLSw+ provides both load balancing (round robin and circuit count) and fault-tolerant options (in both cases, capable secondary peer connections are up all the time). Cisco DLSw+ also supports a concept called backup peers. With backup peers, the peer connections are not activated until the primary connection is disabled. This allows a one-to-n backup structure in very large networks. When using backup peers, if the failed primary peer becomes available again, all new sessions flow over the primary peer. Existing LLC2 sessions (over the backup peer connection) can remain active, terminate immediately, or terminate after a configured amount of time.

IBM 221x DLSw supports the same level of functionality as DLSw+ through the use of DLSw partner neighbor priority. If multiple DLSw partners on the same target LAN are configured with the same neighbor priority, then DLSw will load balance among them. If the DLSw partners are configured with different neighbor priorities, then those DLSw partners with the lower priority will only be used if the ones with the higher priority become unusable (fault tolerance).

Note: In addition to round-robin load balancing, Cisco DLSw+ also supports circuit-count load balancing, where the administrator can specify precisely what proportion of circuits will be brought up over each path. This is useful where multiple peers with different capacities are present.

Cost

Cisco has the ability to assign DLSw peers different costs. When load balancing is not used, the configured cost can be used to choose the preferred DLSw peer. Cisco supports cost from either a local or remote perspective. If cost is specified on a local peer statement, the cost is automatically broadcast to all remote peers, simplifying configuration. Optionally, remote peers can weigh different central site peers with different costs. This allows, for example, the East Coast to prefer one router and the West Coast to prefer another router.

Cisco DLSw+ circuit-count load balancing re-balances the load faster when the preferred-cost peer returns. Without a simple round-robin mechanism, the balance is not restored, since the peer that didn't fail already has circuits and this is not taken into consideration in making the path allocation decision.

The IBM 221x DLSw supports this same functionality through the use of DLSw partner neighbor priority. Essentially, you can configure one of four neighbor priorities (four costs).

Promiscuous and Passive Peers

Promiscuous peering is a Cisco DLSw+ feature that allows the ability to communicate with other DLSw peers that are not pre-configured.

IBM 221x DLSw has different flavors of this capability. First is the ability to enable accepting dynamic neighbor connections from neighbors not configured locally (only one of the two DLSw partners needs to configure the other as a partner). Second is the ability to use DLSw multicast groups to discover all DLSw partners within a group.

Cisco DLSw+ passive-peers is a feature that provides the capability to specify which peer establishes the DLSw connection. A DLSw+ peer configured as passive will wait for the remote peer to initiate a peer connection. This level is especially useful when designing backup scenarios that include Ethernet.

Bandwidth Management and Queuing

Cisco DLSw+ has many methods of managing DLSw traffic within the DLSw partner and within the IP network. Among the queuing methods are priority queuing, custom queuing, and weighted fair queuing. Cisco also has the ability to prioritize types of DLSw traffic within the network using the SNA class-of-service (COS) to IP type of service (TOS) mapping of precedence bits in the IP header.

IBM 221x DLSw supports many equivalent queuing features for DLSw traffic as well. These include the use of circuit priorities within DLSw and BRS outside of DLSw (DLSw V2 does not support TOS queuing).

Local DLSw

Cisco supports the ability to perform a DLC-to-DLC conversion through DLSw+ in one router (without the need to go between DLSw partners). The combinations supported by this local DLSw+ are SDLC-to-token-ring and QLLC-to-LAN.

IBM DLSw V2 also supports this local DLSw feature for a wide range of DLCs. All combinations of DLCs are supported for local DLSw except LAN-to-LAN.

DLSw RFC 1434 Support

IBM's DLSw implementation supports RFC 1434 backward compatibility

Cisco DLSw support does not provide RFC 1434 backward compatibility.

DLSw Multicast

IBM's DLSw implementation supports multicast IP partner discovery, multicast IP group and role membership, and multicast IP resource discovery.

Cisco V2 does not support multicast IP partner discovery, multicast IP group and role membership, or multicast IP resource discovery (these are not part of the DLSw V2 RFC 2166 specification).

Miscellaneous

1. Both Cisco and IBM DLSw implementations support MAC address lists and static cache entries.
2. Both Cisco and IBM DLSw implementations support NetBIOS name lists and static cache entries.
3. Both Cisco and IBM DLSw implementations support SNA/NetBIOS load balancing.
4. IBM DLSw provides SDLC PU2 DLSw dial-in support. Cisco does not support SDLC PU2 DLSw dial-in (it is not part of the DLSw V2 RFC 2166 specification).
5. Both Cisco and IBM DLSw implementations provide SDLC 5394 (or compatible) support.
6. Both Cisco and IBM support remote media conversion (SDLC/QLLC/LLC) for PU4-to-PU4 (Cisco also supports SRB over FDDI, TR LANE, and TR ISL media conversion).

Appendix B. Sample calculations for frame relay parameters

This appendix is provided as an addition, an elaboration, and an explanation of some of the calculations that have to be performed when configuring a frame relay network to be able to transport voice traffic in addition to data traffic.

Consider first of all an environment in which voice calls are being transported over frame relay without encapsulating them in IP; we have referred to this before as VoFR.

If we want to mix voice and data traffic over a 64 kbps[1] PVC and allow for the transport of up to three voice calls simultaneously, we need to ensure that the appropriate number of voice packets can be transmitted during the T_c time period and that data traffic can be fragmented to fill the remainder of the time slot.

For this example, consider voice streams that are encoded at a rate of 9.6 kbps per call. This results in a 25-byte voice frame every 15 milliseconds (once the frame relay header is taken into account) and an effective bandwidth requirement of 13.333 kbps for each voice call. Thus voice can use up to 40 kbps of the 64 kbps bandwidth, restricting data traffic to the remaining 24 kbps at worst.

If T_c is set to 0.03, the minimum value possible (which has to be accomplished by setting B_c to 1920, in fact, and remembering that $B_c = T_c \times CIR$), then in every T_c seconds we will expect a maximum 3 x 25 x 2 bytes of voice traffic. This equates to 150 bytes, or 1,200 bits. Taking this away from B_c we can see that 720 bits of data traffic or 90 bytes can also be transmitted in this time interval. Remember that the transmitting device will police its transmission rate and will not exceed CIR in order to avoid having voice packets discarded by the frame relay network. Since the data traffic will have a six-byte frame relay header, the fragment size should be set to 84 bytes (90 - 6). Under these circumstances, we can therefore transmit six 25-byte voice packets and one 90-byte data packet in one time interval.

If we were to make the fragment size larger than 84 bytes under these circumstances then we run the risk of having voice traffic block data traffic completely. If we are handling voice traffic at the maximum rate of three simultaneous calls, and assuming for now that we always have voice traffic ready to be transmitted, transmission of a data

[1] For the purpose of these calculations we have made the assumption that 64 kbps means 64,000 bits per second and that a byte is exactly eight bits.

fragment larger than 90 bytes will not be allowed because this would exceed the allowed number of bytes in the given time interval. More likely, though, would be the case where we can no longer transmit a voice packet because too much data traffic has already been transmitted in the time interval; either case is to be avoided where possible.

If we increase T_c to 0.06 (by setting B_c to 3840), the calculation changes. Now in the longer time interval we have the ability to send 180 bytes of data traffic, and so the fragment size should now be set to 174 bytes. This is more efficient for transporting data traffic but increases the potential delay to voice packets. Assume that at the beginning of the time interval no voice packets are available for transmission but that data traffic is available for transmission, and furthermore that the frame relay access rate (and hence the speed at which the line is clocked) is 2 Mbps. The router will send 21 data fragments of 180 bytes each and not exceed the committed transmission rate in the time interval. It will take approximately 15 milliseconds to transmit this volume of data traffic at the given access rate. It can then send no more data fragments during this time interval without exceeding the committed rate. Assume that immediately after this data traffic has been transmitted, three 25-byte voice packets are received. To avoid exceeding the contracted rate, only two of these packets can now be transmitted. Furthermore, if another three voice packets arrive after a further 15 milliseconds, none of these can be transmitted during this time interval, but at this point we are only halfway through the 60 millisecond time slot. Hence the trade-off: increasing T_c is more efficient for data traffic but will increase the average delay on the transmission of voice traffic. This does not claim to be a rigorous statistical analysis of the relationship between the T_c value and the average voice delay but some approximate relationship should be clear from the discussion above.

Now consider an example in which voice traffic is initially encapsulated inside IP before being transmitted over frame relay. Assume[2] that each voice circuit is going to require bandwidth for the transmission of 66 bytes of data every 15 milliseconds (20 bytes of voice data, 40 bytes of IP/UDP/RTP headers, 6 bytes of frame relay headers). This approximates to a bandwidth requirement of 35 kbps for each voice call, so we have to assume we can only support a single voice call over a 64 kbps frame relay voice/data circuit.

[2] The exact voice transmission rate depends on the characteristics of the CODEC being used, so the values here should be taken as indicative rather than definitive.

Performing similar calculations as before, T_c=0.03 leads to a data fragment size of 100 bytes and T_c=0.06 leads to a data fragment size of 208 bytes.

Appendix C. Special notices

This publication is intended to help technical specialists plan and implement combined IBM/Cisco router networks. The information in this publication is not intended as the specification of any programming interfaces that are provided by IBM and Cisco. See the PUBLICATIONS section of the IBM Programming Announcement and equivalent Cisco documentation 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, North Castle Drive, Armonk, NY 10504-1785.

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.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

This document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

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:

Advanced Peer-to-Peer Networking
APPN
ESCON
IBM
Netfinity
OS/390
Parallel Sysplex
RS/6000
S/390
SP
SP1
SP2
System/390
VTAM

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere.,The Power To Manage., Anything. Anywhere.,TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET and the SET logo are trademarks owned by SET Secure Electronic Transaction LLC.

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

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 IBM Redbooks

For information on ordering these publications see "How to get IBM Redbooks" on page 173.

- *Inside APPN and HPR - The Essential Guide to New SNA*, SG24-3669
- *Subarea to APPN Migration: HPR and DLUR Implementation*, SG24-5204
- *Application-Driven Networking: Class of Service in IP, Ethernet, and ATM Networks*, SG24-5384

D.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at http://www.ibm.com/redbooks for information about all the CD-ROMs offered, updates and formats.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbooks Collection	SK2T-8038
Lotus Redbooks Collection	SK2T-8039
Tivoli Redbooks Collection	SK2T-8044
AS/400 Redbooks Collection	SK2T-2849
Netfinity Hardware and Software Redbooks Collection	SK2T-8046
RS/6000 Redbooks Collection (BkMgr)	SK2T-8040
RS/6000 Redbooks Collection (PDF Format)	SK2T-8043
Application Development Redbooks Collection	SK2T-8037
IBM Enterprise Storage and Systems Management Solutions	SK3T-3694

D.3 Other resources

These publications are also relevant as further information sources:

- *Systems Network Architecture Advanced Peer-to-Peer Networking, Branch Extender Architecture Reference, Version 1.1*, SV40-0129-01, available only on CD-ROM SK2T-6012

D.4 Referenced Web sites

These Web sites are also relevant as further information sources:

- http://www.ibm.com
- http://www.ibm.com/redbooks
- http://www.cisco.com

How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** http://www.ibm.com/redbooks

  Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

  Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

  Send orders by e-mail including information from the IBM Redbooks fax order form to:

	e-mail address
In United States	usib6fpl@ibmmail.com
Outside North America	Contact information is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Telephone Orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	Country coordinator phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

- **Fax Orders**

United States (toll free)	1-800-445-9269
Canada	1-403-267-4455
Outside North America	Fax phone number is in the “How to Order” section at this site: http://www.elink.ibmlink.ibm.com/pbl/pbl

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at http://w3.itso.ibm.com/ and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at http://w3.ibm.com/ for redbook, residency, and workshop announcements.

IBM Redbooks fax order form

Please send me the following:

Title	**Order Number**	**Quantity**

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.

Index

IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, **please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.**

- Use the online **Contact us** review redbook form found at `http://www.ibm.com/redbooks`
- Fax this form to: USA International Access Code + 1 914 432 8264
- Send your comments in an Internet note to `redbook@us.ibm.com`

Document Number **Redbook Title**	SG24-5865-00 IBM Router Interoperability and Migration Examples
Review	
What other subjects would you like to see IBM Redbooks address?	
Please rate your overall satisfaction:	O Very Good O Good O Average O Poor
Please identify yourself as belonging to one of the following groups:	O Customer O Business Partner O Solution Developer O IBM, Lotus or Tivoli Employee O None of the above
Your email address: The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities.	 O Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.
Questions about IBM's privacy policy?	The following link explains how we protect your personal information. `http://www.ibm.com/privacy/yourprivacy/`

SG24-5865-00
Printed in the U.S.A.

IBM Router Interoperability and Migration Examples

SG24-5865-00