Distributed Routing Software

Routing Protocols Reference Guide

Part Number: AA–QL2CB–TE

December 1995

This guide provides detailed reference information about the micro-operating system structure, and the protocols and interfaces that the bridging routers support.

Revision/Update Information:	This is a revised manual.
Software Version:	Distributed Routing Software V1.0A

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

The following are trademarks of Digital Equipment Corporation: DEC, DECnet, OpenVMS, PATHWORKS, ThinWire, VAX, VAXcluster, VMS, VT, and the DIGITAL logo.

The following are third-party trademarks:

AppleTalk is a registered trademark of Apple Computer, Inc.

Intel is a trademark of Intel Corporation.

IBM is a registered trademark of International Business Machines Corporation.

MS-DOS is a registered trademark of Microsoft Corporation.

NetWare and Novell are registered trademarks of Novell, Inc.

Proteon, ProNET, and TokenVIEW are registered trademarks of Proteon, Inc.

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

This manual was produced by Shared Engineering Services.

Contents

Preface

1 Introduction to Internetworking

2 The Internet Protocol

3 The SNMP Protocol

4 The IPX Protocol

5 The DNA Phase IV Protocol

6 The OSI Protocol

7 The DNA V Protocol

8 The AppleTalk Protocol

9 The Synchronous Data Link Control Relay

10 The X.25 Network Interface

11 The Frame Relay Network Interface

12 The Point-to-Point Protocol

15 The V.25 bis Network Interface

16 The WAN-restoral Interface

17 MAC Filtering

18 NETBIOS Name Caching

19 The DVMRP Protocol

20 Bridging Features

Figures

Tables

Preface

Objectives

Bridging Routers can support a wide variety of network protocols. This guide provides detailed reference information about the micro-operating system structure, and the protocols and interfaces that the bridging routers support.

Audience

This guide is intended for users of the bridging routers, such as network controllers, network administrators and managers, or other related network professionals. If you are an inexperienced user or you are unfamiliar with internetworking concepts, this guide provides a basic understanding of the micro-operating system and the protocols that the bridging router supports. If you are an experienced user, this guide provides detailed current information on the Bridging Router.

Organization

This manual includes the following chapters and a glossary.

- Chapter 1 introduces the concepts and terminology of internetworking with emphasis on where the bridging routers fit into the topology.
- Chapter 2 describes the implementation of the Internet Protocol (IP).
- Chapter 3 describes the implementation of the Simple Network Management Protocol (SNMP).
- Chapter 4 describes the implementation of the IPX™ protocol.

- Chapter 5 describes the implementation of the Digital Network Architecture Phase IV protocol.
- Chapter 6 describes the implementation of the International Organization of Standards (ISO) Open Systems Interconnection (OSI) Connectionless Layer Network Protocol (CLNP).
- Chapter 7 describes the recommendations for running and transitioning to the Digital Network Architecture Phase V protocol.
- Chapter 8 describes the implementation of the AppleTalk protocol, phases 1 and 2.
- Chapter 9 describes the implementation of the Synchronous Data Link Control (SDLC) Relay protocol.
- Chapter 10 describes the implementation of the X.25 interface that connects the router to an X.25 packet-switched network.
- Chapter 11 describes the the implementation of the Frame Relay network interface.
- Chapter 12 describes the implementation of the point-to-point protocol (PPP).
- Chapter 13 describes the implementation of Bandwidth Reservation and Priority Queuing.
- Chapter 14 describes the implementation of the Data Link Switching (DLSw) protocol.
- Chapter 15 describes the V.25*bis* network interface.
- Chapter 16 provides reference information about a Wide Area Network (WAN) restoral interface.
- Chapter 17 explains the MAC Filtering feature available for the bridging router.
- Chapter 18 describes the NETBIOS name caching feature available on the bridging router.
- Chapter 19 describes the DVMRP (Distance Vector Multicast Routing Protocol).
- Chapter 20 describes bridging features that are available with the Adaptive Source Routing Transparent (ASRT) Bridge.
- Glossary. Provides definitions of terminology associated with Bridging Routers and the protocols that they support.

Related Documentation

The following documents provide additional information about the router hardware and software:

- *Bridging Configuration Guide*, AA–QL29B–TE
- *Event Logging System Messages Guide*, AA–QL2AB–TE
- *Network Interface Operations Guide*, AA–QL2BB–TE
- *Routing Protocols User's Guide*, AA–QL2DB–TE
- *System Software Guide*, AA–QL2EB–TE

Conventions Used in This Guide

`Special type` This special type in examples indicates system output or user input.

`Boldface` Boldface type in examples indicates user input.

lowercase-italics Lowercase italics in command syntax or examples indicate variables for which either the user or the system supplies a value.

1

Introduction to Internetworking

This chapter provides an introduction to the concepts and terminology of internetworking with emphasis on where the bridging routers fit into the topology. A general understanding of internetworking is helpful when using bridging routers.

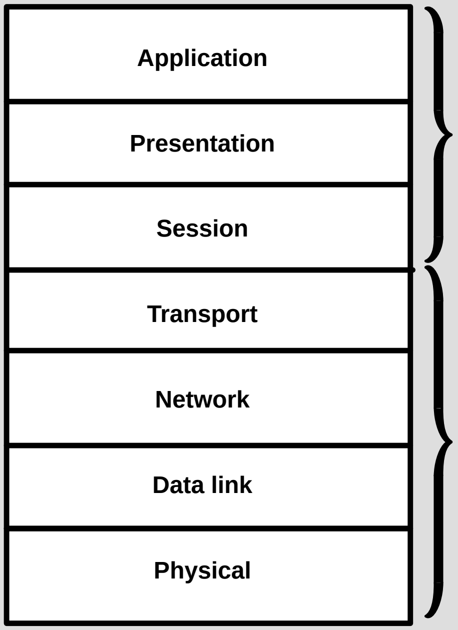
Internetworking

Internetworking connects different networks and LANs through a set of conventions or rules that allow host computers or end systems to communicate with each other. Two organizations, the International Standards Organization (ISO) and the Department of Defense (DoD) each developed a conceptual reference model to govern how host computers or end systems communicate with each other.

OSI Reference Model

The ISO developed the seven layered Open Systems Interconnect (OSI) Reference Model. Figure 1–1 illustrates and describes the services provided by each layer of the OSI Reference Model. A service is a set of functions offered to a user by a provider.

Figure 1–1 OSI Reference Model



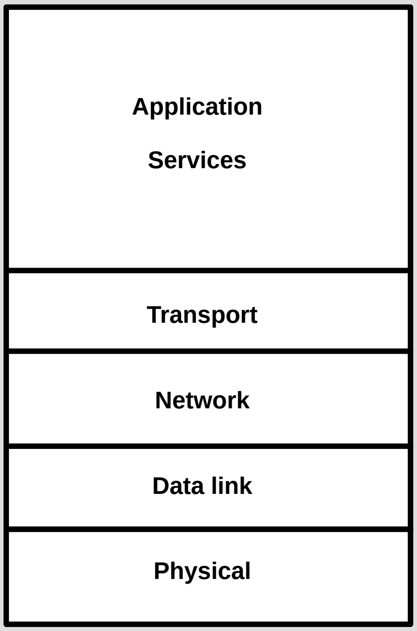
LKG–09878–95I

Layer	Description
Application	Manages communication between application processes.
Presentation	Adds structure to the units of data being exchanged.
Session	Adds control mechanisms to the data being exchanged.
Transport	Reliably transfers the data across the network to the level required by the application.
Network	Transfers data across the network, independent of the media and topology of the underlying segment.
Data Link	Responsible for transmission, framing, and error control of a single communications link.
Physical	Provides specifications for the electromechanical interface to the communications media.

DoD IP Reference Model

The DoD IP reference model has five layers, as compared to the seven layer OSI model. Figure 1–2 illustrates and describes the services provided by the DoD Reference Model.

Figure 1–2 DoD Reference Model



LKG–09878–95I

Layer	Description
Application Services	Manages communication between application processes and adds structure and control mechanisms to the data exchange.
Transport	Reliably transfers the data across the network to the level required by the application.
Network	Transfers data across the network, independent of the media and topology of the underlying segment.
Data Link	Responsible for transmission, framing, and error control of a single communications link.
Physical	Provides specifications for the electromechanical interface to the communications media.

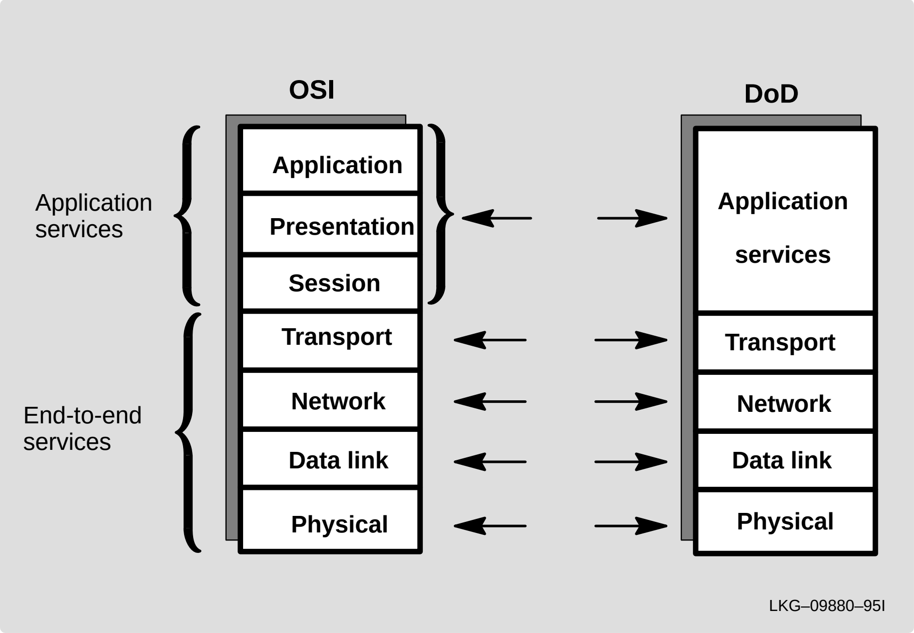
OSI and DoD Comparison

Application services (the upper three layers of the OSI model) are provided by a host computer or an OSI-defined End System (ES). These layers provide all the services necessary for information transfer. Telnet servers for remote consoles is an Application Service provided by bridging routers.

End-to-end services (the bottom four layers of both the OSI and DoD models) provide all the services necessary for data transfer. Bridging routers provide support for the network, data link, and physical layers. Figure 1–3 shows how the OSI and DoD reference models compare to each other.

Note: Throughout the rest of this manual, discussions about the reference model refer to the OSI Model and discussions about services refer to End-to-End services only.

Figure 1–3 OSI and DoD Comparison



Protocol Layering

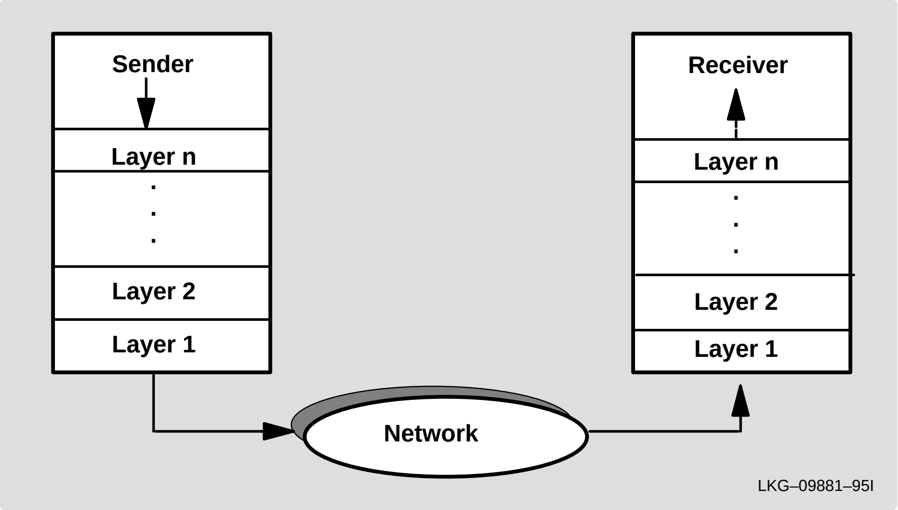
Host computers do not use a single protocol or service for data communication. Instead, host computers require a set of cooperative protocols and services to ensure accurate data transmission from the source to the destination. Services are available through Service Access Points (SAPs). By using a SAP and communicating through a protocol, each service can use services above or below. This concept is referred to as layering. A layer is a simple service that may be augmented to offer more powerful services at the layer immediately above or below. Each layer takes the responsibility for handling one part of the data communication task.

Protocol Stacks

A collection of protocol layers is referred to as a protocol stack. Think of each protocol as being stacked vertically into layers, as in Figure 1–4. Host computers send data from an application program on one machine to an application program on another machine by transferring the message through successive layers of the protocol stack. The source computer transfers data down through the protocol stack, over the network, and up through the protocol stack on the destination computer. The receiver then accepts and uses the data.

In addition to the OSI and DoD protocols, bridging routers also support the following protocols that use protocol stacks: IPX, DECnet Phase IV, and AppleTalk 1 and 2. For more information about these protocol stacks, refer to their respective chapters in this guide.

Figure 1–4 Protocol Layering

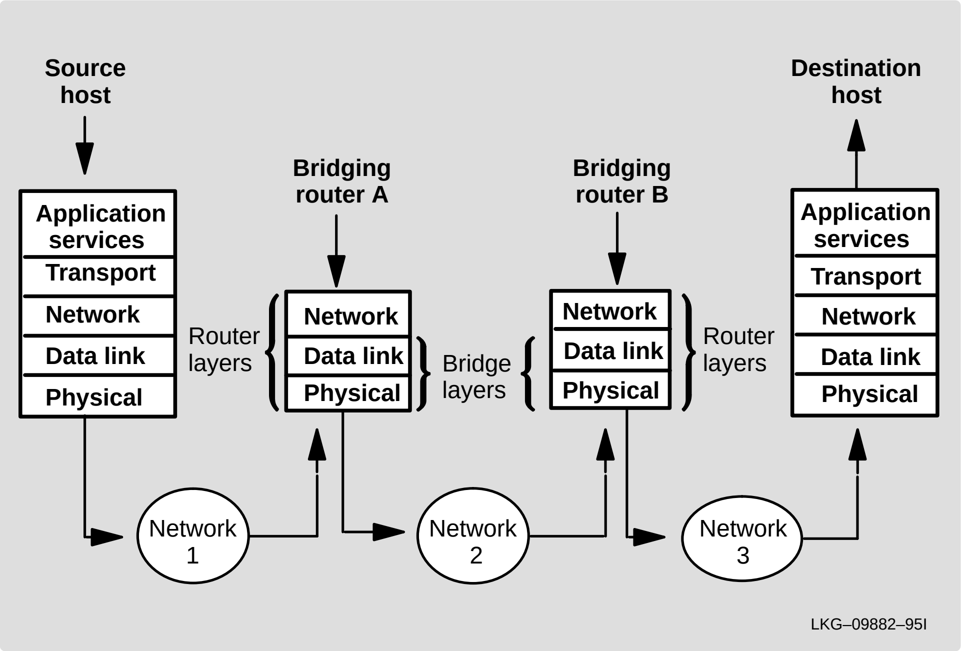


Devices for Connecting LANs and Networks

There are intermediary devices that connect LANs and networks and forward data from the source host to the destination host. These communication devices, bridges, routers, and bridging routers, operate at the Data Link Layer and the Network Layer of the reference models.

Bridges extend or connect similar or dissimilar LANs, while routers connect dissimilar networks. Bridging routers combine the functions of both bridges and routers to provide a comprehensive communications solution. Figure 1–5 shows the host systems and the intermediary communication devices in an internetworking environment. The following sections of this chapter review the functions of bridges, routers, and bridging routers.

Figure 1–5 Network and LAN Connection Devices



Using Bridges to Connect LANs

A *bridge* is a device that transparently connects similar and dissimilar local area networks (LANs). Bridges function at the Data Link layer of the OSI Reference Model. This enables multiple LANs to act as a single entity. Bridging also allows network designers to extend distance restrictions using a technique that is transparent to the user. LANs connected by bridges are often referred to as *segments*. Multiple LANs interconnected are called *extended networks*.

Bridges usually divide LANs into segments that operate separately. As a LAN divider, a bridge selectively isolates a group from its extended network. The bridge then passes only the data specified by the LAN administrator. Most bridges join two segments. However, multiport bridges that join more than two segments are also available.

The bridging routers support two kinds of bridging, Spanning Tree Bridging and Source Routing Bridging. Spanning Tree Bridging maintains a loop-free path between any two end stations. Source Routing Bridging allows an end station to determine the path a packet uses before transmitting the packet.

Spanning Tree Bridging

The Spanning Tree Bridging protocol produces and maintains a loop-free topology in a bridged network that may contain loops in its physical design. In a mesh topology where more than one bridge is connected between two LANs, *looping* occurs when multiple data paths exist between two LANs. In such cases data packets bounce back and forth between two LANs by parallel bridges. This creates a redundancy in data traffic and produces the phenomenon known as looping.

When looping occurs, you must configure the local or remote LAN, or both, to remove the physical loop. With spanning tree, a self-configuring algorithm allows a bridge to be added anywhere in the LAN without creating loops. Upon adding the new bridge, the spanning tree transparently re-configures all bridges on the LAN into a single loop-free *spanning tree.*

Source Routing Bridging

Source Routing Bridging is a non-transparent MAC (Media Access Control) layer bridge (not transparent to the end station) used only on 802.5 LANs. When using source routing, the end station is responsible for determining the route through the bridges and inserting the routing data into the MAC header of the transmitted frames.

Using Routers to Connect Networks

A *router* is a device that connects network segments by selecting a data path through the destination address. One advantage of routers over bridges is that routers function at the Network Layer (layer 3) and bridges function at the Data Link layer. Therefore, only routers have access to network layer addressing information. Routers use this information to determine the optimal path between two end stations. In addition, routers store and forward packets to interconnected network segments regardless of network topology. Routers decide the data path of packets transmitted across networks, based on information in the routing table and the network layer header.

Determining Router Paths

The router accesses the network layer header to retrieve the source and destination network layer address of the packet. The routing table lists the network addresses on the network and the paths between the nodes. The router then uses the information from the network layer header and compares this information to the network addresses in the routing table to determine the best path.

The physical media of the two hosts can be different because the router makes the necessary packet size and addressing scheme adjustments.

Address Resolution Protocol

The Address Resolution Protocol (ARP) is a low-level protocol that dynamically maps network layer addresses to physical medium access control (MAC) addresses. ARP allows the source host or router, to find the MAC address of the destination host on the same network segment, given only the network layer address of the destination system.

For example, a router receives an IP packet destined for a host connected to one of its LANs. The packet only contains a 32-bit IP destination address. To construct the data link layer header, a router must acquire the physical MAC address of the destination host and map that address to the 32-bit IP address. This function is called *address resolution.*

ARP Address Translation

When a router translates a network layer address to a physical address, first the router accesses the ARP (translation) cache for the physical MAC address that corresponds to that network layer address. If the cache does not contain the physical MAC address, then the router broadcasts an ARP request to all hosts requesting a response from the host with the correct physical MAC address. The destination host with the correct physical MAC address sends a positive response to the router. The router sends the packet to the destination host and enters the physical MAC address into the translation cache for future use. Figure 1–6 illustrates how ARP works.

ARP Subnet Routing

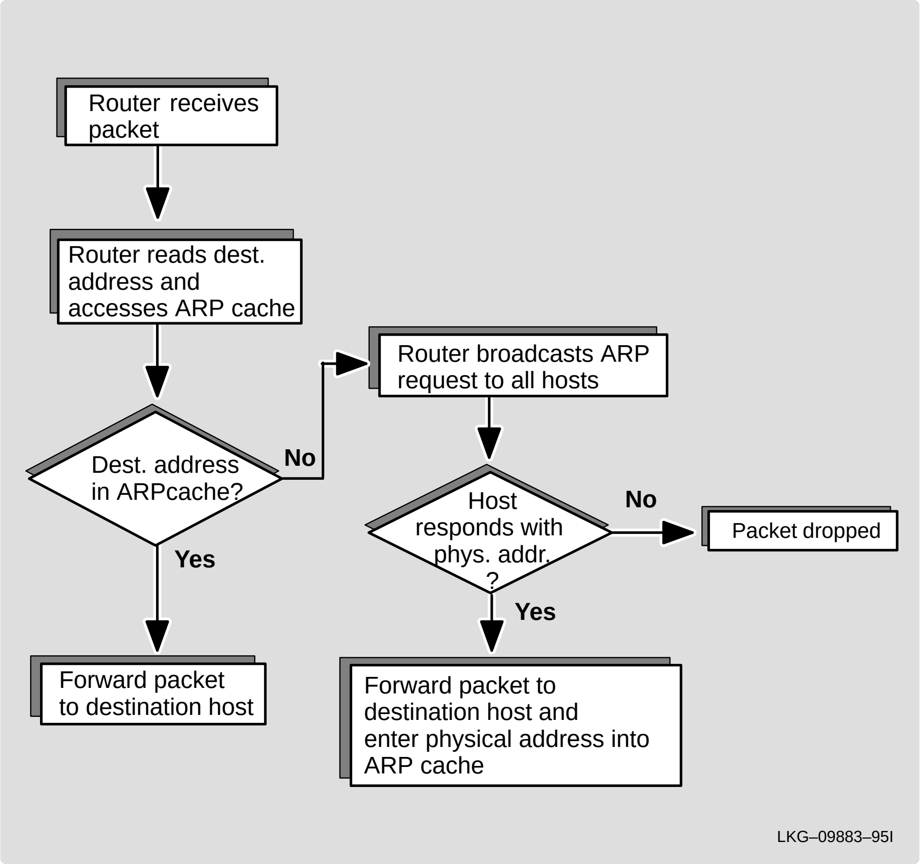
ARP subnet routing is also called *Proxy ARP* and *Promiscuous ARP.* ARP subnet routing deals with hosts that do not support IP subnet routing, but have an interface on a subnetted network. This subnet network must use ARP to translate the IP address to hardware address.

If the host attempts to send a packet to a network subnet, it sends an ARP request to find the hardware address of the destination host. If the subnet is not on the local wire, a router configured for ARP subnet routing may respond to the ARP request with its own hardware address if the following conditions exist:

- The router has the location of the subnet in its routing table.
- The router sends packets to that subnet through a different interface from the interface that received the ARP request.

Because of the second condition, you should configure all routers on a local wire for ARP subnet routing when you use hosts without network subnet support.

Figure 1–6 ARP Physical Address Broadcast


Router receives packet
Router reads dest. address and accesses ARP cache
Dest. address in ARPcache?
No
Yes
Router broadcasts ARP request to all hosts
Host responds with phys. addr. ?
No
Packet dropped
Yes
Forward packet to destination host
Forward packet to destination host and enter physical address into ARP cache
LKG–09883–95I

Using Bridging Routers to Connect LANs and Networks

A *bridging router* is a device in which both bridge and router software run concurrently and operate in the following manner:

- Packets are routed if a specific routing protocol is globally enabled.
- Packets are filtered if you have configured the specific protocol filters.
- Packets are bridged if they are not routed or filtered. In this case, they are forwarded according to their destination MAC address.

Note: Source routing uses the features of both routing and bridging.

Bridges work at the data link layer providing connectionless service. Routers operate at the network layer and therefore offer some flexibility in choosing either connection-oriented or connectionless service. Source routing bridges provide a connection oriented service.

Bridging routers forward packets over networks in the following way:

1. The source host builds a packet from the application layer downward to the physical layer and passes the packet over the network to a bridge or router.
2. If a bridge receives the packet, the bridge accesses the destination MAC address that is stored at the data link layer.

   If a router receives the packet, the router accesses the network layer that contains the destination network address of the packet.
3. In a Spanning Tree Bridging environment, the source and destination MAC address are compared to a list of addresses contained in the source and destination address filtering tables on the bridge to obtain a match. If a match is obtained, the packet is dropped.

   In a Source Routing Bridging environment, the bridge accesses the address and the path to reach the destination address from response to the router discovery frame.

   In a routing environment, the router compares the network layer destination address to the addresses in the routing table to obtain the best match. If the best match is obtained, the router rebuilds the data link layer and passes the packet to the next hop.
4. If the sender forwards the packet to a host, the host transmits an acknowledgment to the sender, depending on the type of transport layer service the host uses.

2

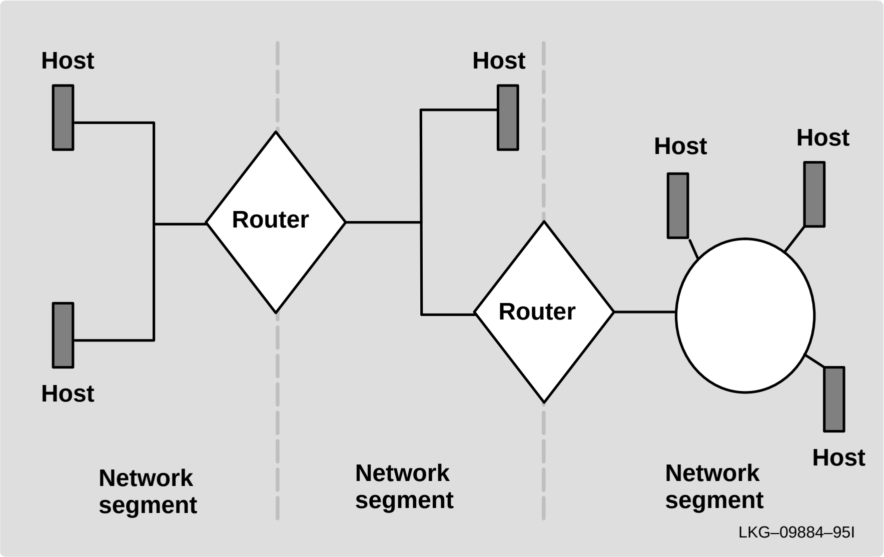
The Internet Protocol

This chapter describes the Internet Protocol (IP) implementation.

IP Network Overview

An *IP network* is a group of network segments that are interconnected by devices called routers. Each segment consists of a router and a host or groups of hosts that use a particular IP network number. The segments and hosts for a particular IP network can be thought of as a *virtual network* (Figure 2–1).

Figure 2–1 IP Network



IP is a non-proprietary network layer protocol that provides connectionless service for the delivery of packets. IP makes a best effort to deliver packets to destinations with no guarantee that the packet arrives. Reliability is left to the transport layer protocol such as TCP (Transmission Control Protocol).

This IP implementation conforms to standards defined by the TCP/IP protocol suite.

IP Addressing

IP addresses identify a host on a particular IP network. If, for example, a host has more than one interface attached to the network, that host has an IP address for each connection (the address is associated with the host). This makes an IP address much like a post office box recipient, who receives the mail whether it arrives on foot or by truck.

IP Address Hierarchy

An IP address is a 32-bit number used in the header of an IP datagram that encodes network segment identification as well as identification of a host on that network. These 32-bit numbers are commonly represented in dotted decimal notation: a decimal integer represents one octet of the 32-bit address.

Thus a 32-bit IP address

```
10000000   00001010   0000010   00011110
```

is written

```
128.10.2.30
```

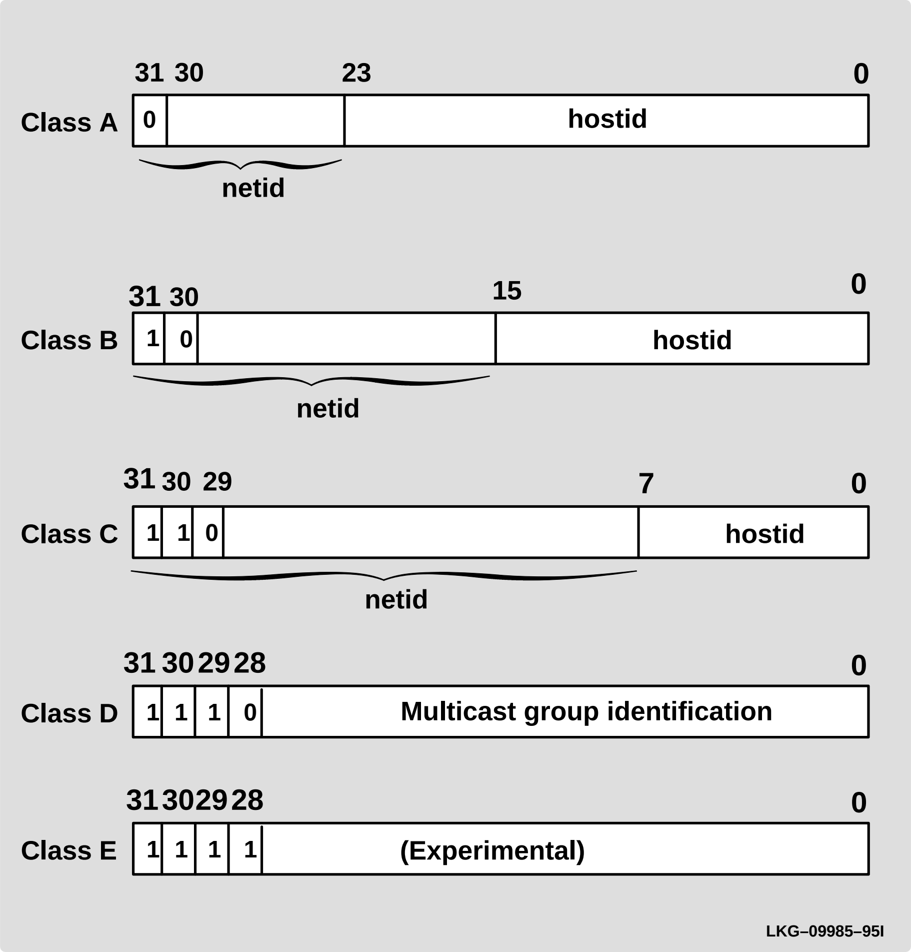
Each IP address forms a pair of identifiers, the *netid* and the *hostid.*

- The netid identifies the network.
- The hostid identifies a host on that network.

IP addresses have three primary forms of designation: Class A, Class B, Class C, Class D, and Class E (experimental). A host determines the class of IP address by examining the high order bits of the address.

Network IP addresses are 32-bit entities that are expressed in Network Byte order. Figure 2–2 illustrates the IP address classifications.

Figure 2–2 IP Address Classifications



Note: Throughout the remainder of this chapter, dotted decimal notation is used to express IP addresses.

Class A Address

A Class A address is used for any network having more than 65,536 hosts. Only 127 Class A network numbers exist.

A host interprets a Class A address by reading bit 31 of the 32-bit address. If this bit is set to 0, the host interprets the netid as the first 8 bits (most significant bits) and hostid as the last 24 bits (least significant bits).

Class B Addresses

A Class B address is used for any intermediate size network having between 256 and 65,536 hosts. With this address the first 16 bits of the 32-bit address are devoted to the netid and last 16 bits are devoted to the hostid.

A host interprets this address by reading bits 31 and 30 of the 32-bit address. If these bits are set to 1 and 0 respectively, then the host interprets the netid field as the first 16 bits (most significant bits) and the hostid field as the last 16 bits.

Class C Addresses

A Class C address is used for any network having fewer than 256 hosts. With this address the first 24 bits are devoted to the netid and last 8 bits to the hostid.

A host interprets this address by reading bits 31, 30, and 29 of the 32-bit address. If these bits are set to 1, 1, and 0, respectively, then the host interprets the netid field as the first 24 bits and the hostid field as the last 8 bits.

Class D Addresses

A Class D address is used for IP multicasting. With this address bits 31, 30, 29, and 28 contain 1, 1, 1, and 0, respectively, and identify the address as a multicast. Bits 27 through 0 identify the specific multicast group.

Class E Addresses

A Class E address (experimental) is indicated when bits 31, 30, 29, and 28 are all set to 1.

Multiple IP Addresses (Same Interface)

This implementation of IP allows you to assign multiple IP addresses on the same interface. Multiple IP addresses allow flexibility when

- Migrating from one IP address to another.

- Using multiple subnets or multiple IP networks on the same physical network segment. For example, it is possible that the number of hosts on the physical network segment exceeds the current subnet's capacity. When this occurs, another subnet must be added to the physical network segment.

Note: When using multiple IP addresses, make sure that each host can accept the IP broadcast address that the network is using. The correct IP broadcast address is all 1's. However, due to BSD implementation's wide distribution, a host ID of all 0's may be used also. Some use 255.255.0.0; others use 0xFF.0xFF.0.0.

Subnet Addresses (Subnetting)

The concept of subnet addressing or subnetting allows a site to partition its network with routers (for example, for security, or work groups, or physical connections) without obtaining multiple IP net addresses. Subnetting adds another level of hierarchy to the internet addressing structure. Instead of a 2-level (netid, hostid) hierarchy, there is now a 3-level (netid, subnetid, hostid) hierarchy. An organization is then free to assign a distinct subnet number and configure its internal IP network as it pleases.

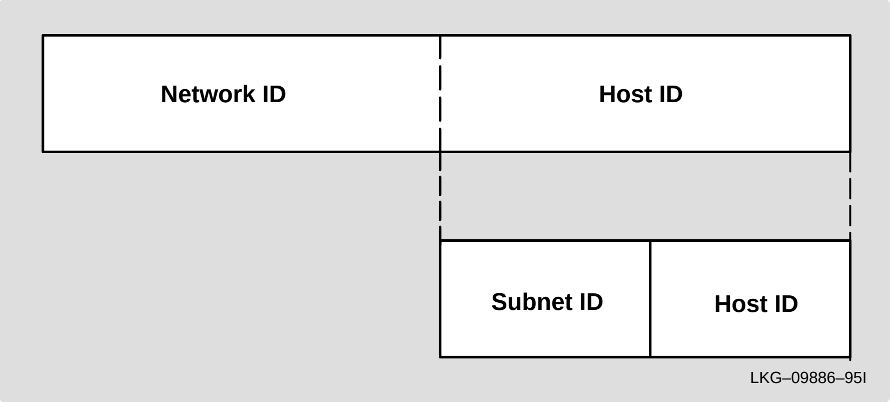
An organization's subnet structure is never visible outside the organization's network from a host (or router) located anywhere else on the internet. It gives the organization the freedom to administer its internal network without having to deal with internet management.

Conceptually, adding subnetting only changes the interpretation of the IP address. Subnetting divides the address into a network ID, subnet ID, and host ID. The network segment is then identified by a combination of network ID and subnet ID.

A subnet mask must contain at least 2 bits not used in the standard network mask but must not contain more bits than 2 less than the standard number of bits in the host ID. Any bits in the host ID bits may be used for the subnet mask. These bits do not need to be contiguous the the network mask or to themselves. For example, the net mask and subnet mask for some Class B network can be 255.255.85.170 (0xFF.0xFF.0x55.0xAA).

Figure 2–3 shows the subnet addressing concept.

Figure 2–3 Subnet Concept

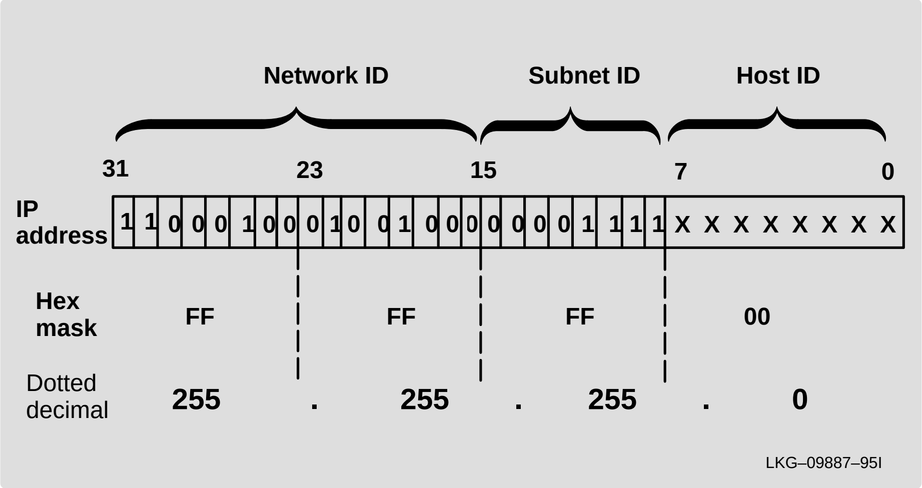


Subnet Mask

When adding an IP address to an interface, you must specify the subnet mask. Subnet masks identify the portion of the address occupied by the netid and the subnetid. The mask is simply another 32-bit string written in dotted decimal notation with all ones in the network and subnet portion of the address. For example, suppose you have a Class B address. You want to assign the most significant 8 bits of the hostid as the subnetid. Following the rule of placing all ones in the netid and subnetid fields, you get a mask of 255.255.255.0. Another example includes using the most significant 2 bits of the host ID, which gives an IP address of 255.255.192.0, and results in 2^{14} possible host IDs.

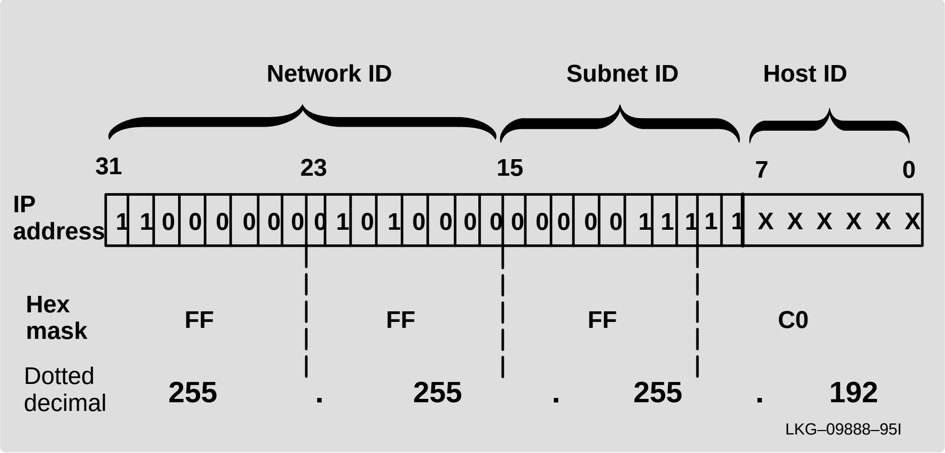
Figure 2–4 describes the 8-bit subnet mask.

Figure 2–4 8-bit Subnet Mask



The subnetid can consist of any number of host field bits; they do not have to be multiples of eight. For example, you may want to assign the first ten bits of the hostid as the subnetid. This creates a mask of 255.255.255.192. Figure 2–5 illustrates this mask.

Figure 2–5 10-bit Subnet Mask



Use two or more bits for subnetid; a subnetid of two bits yields only four subnets, two of which are reserved (the 1,1 and 0,0 values). A two bit subnet mask sometimes offers a good way to partition in half.

Table 2–1 shows the subnet masks subnet and host fields that you can get from dividing an octet.

Table 2–1 Subnet Masks

Subnet Bits	Host Bits	Byte of Hex Mask	Byte of Decimal Mask
0	8	0	0
1	7	0x80	128
2	6	0xC0	192
3	5	0xE0	224
4	4	0xF0	240
5	3	0xF8	248
6	2	0xFC	252
7	1	0xFE	254
8	0	0xFF	255

Note: All interfaces on the same IP network must use the same subnet mask if RIP is being used across the interfaces.

Variable-Length Subnets

This IP implementation also supports variable-length subnets. This feature allows you to divide the hostid of a single IP network number into many variable size subnets. For example, 128.185 could have a subnet 128.185.22.0 with subnet mask of 255.255.255.0 (giving a size of 254 possible hosts) and a subnet 128.185.23.16 with mask 255.255.255.240 (giving a size of 14 possible hosts).

Variable-length subnets can only be used with dynamic routing protocols that distribute each destination’s subnet mask (for example, the OSPF and Integrated IS-IS routing protocols). The IP protocol also allows static routing information to take advantage of variable-length subnetting.

Caution: Assign variable-length subnets with care. If you a assign a subnet in an overlapping fashion, unexpected and possibly incorrect routing may occur.

Obtaining an IP Address

If you are planning to connect your networks and routers to the TCP/IP Internet maintained by DARPA (Defense Advanced Research Projects Agency), you must get registered IP network addresses from the addressing authority at the Stanford Research Institute's Network Information Center.

Autonomous Systems

The other main facet of IP addressing is the Autonomous Systems (AS). In a large internet such as the one described above, no single administrative authority has control of the whole system. Rather, the system consists of many cooperating groups. Often each group wants to retain a high degree of control over its piece of the system to prevent contamination from problems elsewhere in the system.

For this purpose, groups of routers are arranged into autonomous systems. ASs are numbered sequentially by 16-bit identifiers with no structure. A network addressing authority gives out these identifiers. There is no direct connection between IP addresses and AS numbers. Typically, an organization is assigned a single AS number.

The following section on IP Routing further describes the function of autonomous systems.

IP Routing

IP uses routing tables to decide where to send a packet. The routing table is a list of all the IP networks that IP knows how to reach. The routing table contains both dynamic and static routes.

A *dynamic* route is one that is learned through Integrated IS-IS, OSPF, RIP, or EGP. These protocols regularly update their routing tables as network conditions change. Dynamic routing allows the router to transmit packets around network failures.

A *static* route is a route that never changes and that you must enter when configuring IP. Static routes persist across power-downs, restarts, and software reloads. They are used when the router for some reason cannot determine the correct route dynamically. IP routing happens as follows:

1. The destination MAC address must be one of the MAC addresses used by the brouter.

2. IP receives the packet and reads the 32-bit destination address found within the packet header.

3. If the destination IP address is destined for this router, further routing is not necessary and this packet is processed as if by an end system (with the exception of certain ICMP packets). Packets in this category include the following:

   – Control packets for IP itself (ICMP)

   – Routing update packets

   – Packets used for diagnostic purposes

   – SNMP

   – Telnet

4. If the packet is destined for a host on a directly-connected IP network, IP matches the 32-bit destination address with the appropriate physical address in the ARP table. IP then hands the packet to the appropriate lower-level protocol module for transmission directly to the destination node.

5. If the packet is destined for a host on a remote IP network, IP uses the routing table to determine which router leads to that network segment. Each entry in the routing table contains a destination address – in the form an IP network address (optionally with the subnet for the local IP address) or IP host address (32-bit: net ID and host ID) – and the IP address of the next hop router. If IP matches the destination address in the table with the destination contained in the packet, the packet is handed to the appropriate lower-level protocol module for transmission to that next hop.

6. If the packet has no entry for its IP address in the routing table, the packet is routed to the default router (if one is defined or learned). Default routers are used to route packets whose destination address is not found in the routing table. This router is assumed to know the location of the packet's destination. Refer to the Default Router section for more information.

IP also performs several other major tasks:

- Maintaining default routers.
- Martian Filtering.
- Using unnumbered serial lines for network traffic.
- Access Control – You can control access of packet traffic to IP networks, subnets, and hosts on those nets and subnets.

Default Routers (Gateways)

A default router knows how to route packets that other routers cannot route. There are two kinds of default routers: Default Network and Default Subnet.

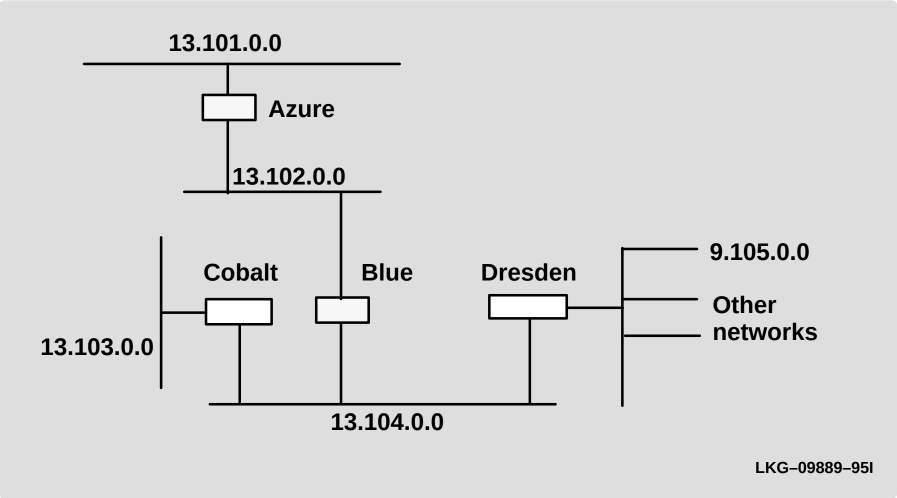
- Default Network routers perform routing for other routers on an internet that has packet traffic for an unknown-network destination.
- Default Subnet routers perform subnet routing in a network where the other routers do not know how to route traffic for specific subnets.

You can specify a route (static route) where IP routes packets that it cannot route using its own routing table; or, the router can learn about the default gateway by using the OSPF, Integrated IS-IS, or RIP protocols. These protocols represent the default route as destination 0.0.0.0 (with a mask of 0.0.0.0).

In Figure 2–6, the network segments are 13.101.0.0, 13.102.0.0, 13.103.0.0, 13.104.0.0, and 9.105.0.0.   The routers are Azure, Blue, Cobalt, and Dresden, where Dresden is the default network gateway because it has knowledge of network 13 and any other networks. Network 13 routers do not have any knowledge of networks outside network 13.

On network segment 13.104, unknown-network traffic goes first to router Dresden then toward the appropriate destination.

Figure 2–6 Internet with Default Gateway (Dresden)



Martian Filtering

Martian is a term that applies to packets that are incorrectly formatted or have an improper destination address. The router drops these packets to ensure that Martians are not forwarded further into the network.

Address filtering ensures that the IP does not forward any packets to those specified addresses, nor does it broadcast any routing information it receives concerning those addresses. For example, address 127.0.0.0 is used as a local loop-back address in BSD-derived UNIX-based operating systems. It is recommended that you install a filter to prevent ill-behaved host systems from transmitting packets destined for the loopback to be transmitted across your internetwork.

Unnumbered Serial Lines

This implementation of IP allows you to send IP traffic over a serial line interface without assigning an IP network number to that line. This feature allows you to configure static routes across the line to the next hop router or to a default router. This feature also includes the following restrictions on certain diagnostic capabilities:

- You cannot PING the interface to find out if it is functional.

- The RIP protocol does not send or transmit subnet routes over unnumbered serial lines.

Access Control

This feature allows you to control the forwarding of packets by examining the masked source and masked destination addresses in the IP header, the protocol type in the IP header, or the port number in the TCP or UDP headers.

After enabling access control, any packet that the router receives is matched to the access control list before being matched to the routing table. Depending on the type of entry on the control list, the packet may be forwarded or may be dropped.

There are two types of entries in the access control list: inclusive and exclusive. If an address matches an inclusive entry, the packet is forwarded. If an address matches an exclusive entry, the packet is dropped. If no match exists, the packet is also dropped.

Use caution when using access controls. Packets originated by the router are also subjected to access controls before being forwarded.

Note: Do not filter out any RIP or OSPF packets being sent or received by the router. You can use the wildcard inclusive entry as the last entry in the access control list, or explicitly include them.

Router ID

The router ID is the *default IP address* that the IP protocol uses when sourcing various kinds of IP traffic. When the router ID is set:

- The router ID becomes the source IP address in all locally-originated IP packets that are either sent over unnumbered serial lines or are multicast.

- The router ID is used as the OSPF router ID.

Internal IP Address

The internal IP address is an address that belongs to the router as a whole, and not any particular interface. It is used only in situations where the router needs to have a particular address available, or when the router is the source of many multicast packets. (For example, when the router is running IP multicast bridging tunnel.)

When the internal IP address is set:

- The internal IP address becomes the source IP address in all locally-originated IP packets that are either sent over unnumbered serial lines or are multicast.
- The internal IP address is used as the OSPF router ID.
- The internal IP address is **not** equal to any of the interface addresses, it is advertised as a host route in the OSPF router-LSA.
- To 0.0.0.0, the internal IP address is deleted.
- The router ID is also set, the internal IP address takes precedence over the router ID.

Note: It is recommended that you only use the internal IP address when there is a need to have an IP address that is always reachable.

Management Applications Module Address for DEChub 900

The Management Applications Module (MAM) is a hub component. It acts as an interface between the hub and the attached modules to provide IP services through the hub.

The DEChub 900 requires modules attached to the hub that have MAC addressing capability to provide proxy end system functionality for IP services. To support IP services such as SNMP, Telnet, and other IP protocols, the router must provide end system network layer addressing.

You must assign the IP network address for the MAM as part of one of the network interfaces configured for the router. The MAM host number must be different from the host number assigned to the router interface that shares that network. The host selection for the MAM and the router must not conflict with any attached host on the shared network.

Broadcast Packets

A broadcast message is one that is destined for all hosts on the given network. IP occasionally sends broadcast addresses on its own behalf. For example, these broadcast messages are used to update the IP routing tables on other routers when running RIP. It is generally considered bad practice to forward broadcast packets or respond to them in any way, and the routers follow these guidelines.

Note: When configuring the router's broadcast address, it is best if all nodes or systems on the wire use the same broadcast format.

To indicate that a packet is a broadcast packet (intended for all hosts), the sender sets the packet's IP destination address to the currently used broadcast address. The broadcast style that you configure is either a local-wire broadcast or network broadcast that uses a fill pattern of all ones or all zeros. During a local-wire broadcast the entire destination address is filled with the pattern. During a network broadcast, only the hostid is filled with the pattern.

Table 2–2 lists the local and network broadcast fill patterns.

Table 2–2 Broadcast Fill Patterns

Broadcast Type	Broadcast Pattern		Hex Example	Dotted Decimal Example
Local Wire	all 0s	N/A	00 00 00 00	0.0.0.0
Local Wire	all 1s	N/A	FF FF FF FF	255.255.255.255
Network	N/A	Class A: all 0s all 1s	12 00 00 00 12 FF FF FF	18.0.0.0 18.255.255.255
Network	N/A	Class B: all 0s all 1s	8E 14 00 00 8E 14 FF FF	143.20.0.0 143.20.255.255
Network	N/A	Class C: all 0s all 1s	C8 29 03 00 C8 29 03 FF	200.41.3.0 200.41.3.255

Note: Network-style broadcast messages include the network and subnet number of the network where they are destined. The IP requirements specify all ones (binary) for the fill pattern in broadcast addresses. BSD 4.2 UNIX requires all zeros.

Receiving IP Broadcasts

The IP protocol recognizes all forms of broadcast messages and addressing. If the network portion of the broadcast address indicates either *local wire* or a *directly-connected* IP network, IP treats the packet as if it is addressed to itself.

IP also forwards directed broadcasts. A *directed broadcast* is a broadcast destined for networks other than the network on which it originated. By enabling IP's directed broadcast feature, you can forward IP packets whose destination is a non-local (remote LAN) broadcast address. For example, a packet originated by the source host is unicast.   This packet is then forwarded, as a unicast, to a destination subnet and *exploded* into a broadcast. You can use this feature to locate network servers and to enable both the forwarding and exploding of directed broadcasts. The default setting for the directed broadcast feature is enabled.

IP Multicast Routing

The IP protocol also supports IP multicast routing through IP multicast extensions to OSPF (MOSPF).   This functionality is explained in greater detail in the section "The OSPF Routing Protocol" found later in this chapter.

IP Multicast Applications

Many applications that make use of multicasting on a single LAN can also take advantage of IP multicast support. Sample applications include distributed computing, voice and video conferences, replicated databases, and resource location. While IP multicast capability is not yet widespread, a number of TCP/IP applications are specified with "hooks" for future multicast enhancement. Examples of these applications include NetBIOS over TCP/IP and the tunneling of IPX traffic through IP networks.

The following existing TCP/IP applications take advantage of the multicast support provided by this MOSPF implementation:

- Any application that uses the IGMP protocol to establish group membership. Examples of this include the Silicon Graphics' Dogfight program and the voice conferencing program ("vat") that is run over DARTNet. Both of these applications run over UNIX and require a multicast kernel. They also usually use DVMRP as their multicast routing protocol, but MOSPF can be substituted for DVMRP without loss of functionality.

- The router's IP console supports a **ping** command that accepts a class D address as destination. The **ping** command displays the IP addresses of the (possible) multiple responders.

- The router’s OSPF console supports **join** and **leave** commands that enable the router to establish multicast group membership. After joining a multicast group, the router responds to pings and SNMP queries sent to the group address. If you want to make the router’s group membership permanent in the configuration, you can use the analogous **join** and **leave** commands in the OSPF configuration console.

Tunneling Other Network Protocols Over IP

To allow IBM terminal traffic and IBM LAN traffic to merge with non-IBM traffic across a single backbone, the Source Routing Bridge Tunnel and SDLC (Synchronous Data Link Control) Relay features of the bridging router software *encapsulate* IBM traffic within industry-standard TCP/IP packets. The bridging router then routes these packets through an IP path or *tunnel*, allowing you to benefit from increased functionality and network utilization as well as higher network availability and increased ease of use.

End stations see the IP path (the tunnel) as a single hop, regardless of the network complexity. This helps overcome the usual 7-hop distance limit encountered in source routing configurations. It also lets you connect source routing end stations across non-source routing media, such as Ethernet networks.

The bridging tunnel also overcomes several limitations of regular source routing including the following:

- Distance limitations of seven hops

- Large amounts of overhead that source routing causes in wide area networks (WANs)

- Source Routing’s sensitivity to WAN faults and failures (if a path fails, all systems must restart their transmissions)

With the bridge tunnel feature enabled, the software encapsulates packets in TCP/IP packets. To the router, the packet looks like a TCP/IP packet. Once a frame is encapsulated in an IP envelope, the IP forwarder is responsible for selecting the appropriate network interface based on the destination IP address. This packet can be routed dynamically through large internetworks without degradation or network size restrictions. End stations see this path or tunnel, as a single hop, regardless of the complexity of the internetwork.

The tunnel is transparent to the end stations. The bridging routers participating in tunneling treat the IP internet as one of the bridge segments. When the packet reaches the destination interface, the TCP/IP headers are automatically removed and the inner packet proceeds as a standard source routing packet.

Encapsulation and OSPF

A major benefit of the encapsulation feature is the addition of the OSPF or Integrated IS-IS dynamic routing protocols to the routing process. These protocols offer the following benefits when used with encapsulation:

- **Least-Cost Routing** – These protocols access the fastest path (tunnel) with the fewest delays allowing network administrators to distribute traffic over the least expensive route.

- **Dynamic Routing** – These protocols look for the least-cost path as well as detects failures and reroutes traffic with low overhead.

- **Multi-Path Routing** – Load sharing makes more efficient use of available bandwidth.

With OSPF or Integrated IS-IS, tunnels automatically manage paths inside the internetwork. If a line or bridge fails along the path then the tunnel bridge automatically reroutes traffic along a new path. If a path is restored, the tunnel automatically updates to the best path. This rerouting is completely transparent to the end stations.

MOSPF Tunnels

The current version of router software limits the amount of spanning tree explorer frame traffic (or other packets that are broadcast by the bridge) that is forwarded over a tunnel. This feature allows you to save on WAN bandwidth.

You can form subsets of tunnel endpoints into groups. The groups are defined by a number between 1 and 64 and can be either peer groups or client/server groups.

When a router receives a spanning tree explorer frame, it associates the explorer with a group through the universal filter's tagging mechanism. If the frame belongs to a peer group it is sent to all the other members of the group. If the explorer belongs to a client/server group it is sent to all of the group's servers or clients (depending on whether the router is labeled as a client or a server, respectively, for the given group). Explorers that are not labeled are either sent to a special group (Group 0) or to all tunnel endpoints.

See the "OSPF and IP Multicast Routing" section in this chapter for more information about MOSPF. Refer to the *Bridging Configuration Guide* for more information about bridging and tunnels.

Synchronous Data Link Control (SDLC) Relay

The SDLC Relay feature consolidates serial lines by combining SNA SDLC and LAN networks. This feature allows point-to-point SDLC transmission between SNA devices by encapsulating SDLC frames in industry-standard TCP/IP packets and using dynamic routing. SDLC relay provides SDLC frame transmission across an internetwork, providing predictable response time with minimal protocol overhead. For more information about this feature, refer to the chapter "The Synchronous Data Link Control Relay."

Interior Gateway Protocols

Routers that use a common routing protocol form an *autonomous system* (AS). This common routing protocol is called an Interior Gateway Protocol (IGP). IGPs dynamically detect network reachability and routing information within an AS and use this information to build the IP routing table. IGPs can also import external routing information into the AS.

The router supports three different IGPs for building the IP routing table: OSPF, RIP, and OSI's Integrated IS-IS. OSPF and Integrated IS-IS are based on link-state technology or the shortest-path first (SPF) algorithm. RIP is based on the Bellman-Ford or the distance-vector algorithm.

The routers can simultaneously run OSPF and RIP or Integrated IS-IS and RIP. In general, use of the OSPF or Integrated IS-IS protocols is recommended due to their robustness, responsiveness, and decreased bandwidth requirements.

Note: The OSPF technical description is covered in the next section to maintain continuity when describing the different IGPs that IP supports; however, when configuring the IGPs, OSPF is configured separately within the CONFIG environment.

Integrated IS-IS is used in a dual routing domain, where there is a need to route both IP and OSI traffic. When using Integrated IS-IS, OSPF cannot be used.

The OSPF Routing Protocol

The router supports a complete implementation of the OSPF routing protocol, as specified in RFC 1247 (Version 2). This version is incompatible with bridging routers running OSPF version 1. OSPF information is not be exchanged between routers running version 1 and version 2.

OSPF is a link state dynamic routing protocol that detects and learns the best routes to (reachable) destinations. OSPF can quickly perceive changes in the topology of an AS, and after a short convergence period, calculate new routes.

OSPF is designed to provide services not available with RIP. OSPF features include the following:

- **Least Cost Routing** – Allows you to configure path costs based on any combination of network parameters. For example, bandwidth, delay, and dollar cost.

- **No limitations to the routing metric** – While RIP restricts the routing metric to 16 hops, OSPF has no restriction.

- **Multipath Routing** – Allows you to use multiple paths of equal cost that connect the same points. You can then use these paths for load balancing resulting in more efficient use of network bandwidth.

- **Area Routing** – Decreases the resources (memory and network bandwidth) consumed by the protocol and provides an additional level of routing protection.

- **Variable Length Subnet Masks** – Allow you to break an IP address into variable size subnets, conserving IP address space.

- **Routing Authentication** – Provides additional routing security.

OSPF Routing Domain

Each router running the OSPF protocol has a database describing a map of the routing domain. This database is identical in all participating routers. From this database the IP routing table is built through the construction of a shortest-path tree, with the router itself as root. The routing domain refers to an AS running the OSPF protocol.

OSPF supports the following physical network types:

- **Point-to-Point** – Networks that use a communication line to join a single pair of routers. A 56 Kb serial line that connects two routers is an example of a point-to-point network.

- **Broadcast** – Networks that support more than two attached routers and are capable of addressing a single physical message to all attached routers. Token-ring, Ethernet, and FDDI networks are examples of broadcast networks.

- **Non-Broadcast** – Networks that support more than two attached routers but have no broadcast capabilities. An X.25 Public Data Network is an example of a non-broadcast network. For OSPF to function properly, this network requires extra configuration information about other OSPF routers attached to the non-broadcast network.

OSPF Areas

OSPF allows you to split the AS into regions called areas. OSPF areas are a collection of contiguous networks. The topology of any one area is hidden from that of the other areas. Hiding information significantly reduces routing traffic and protects routing within an area from outside influence.

A router has a separate database that contains the topology for each area to which it is connected. Two routers belonging to the same area have identical topologies for that area.

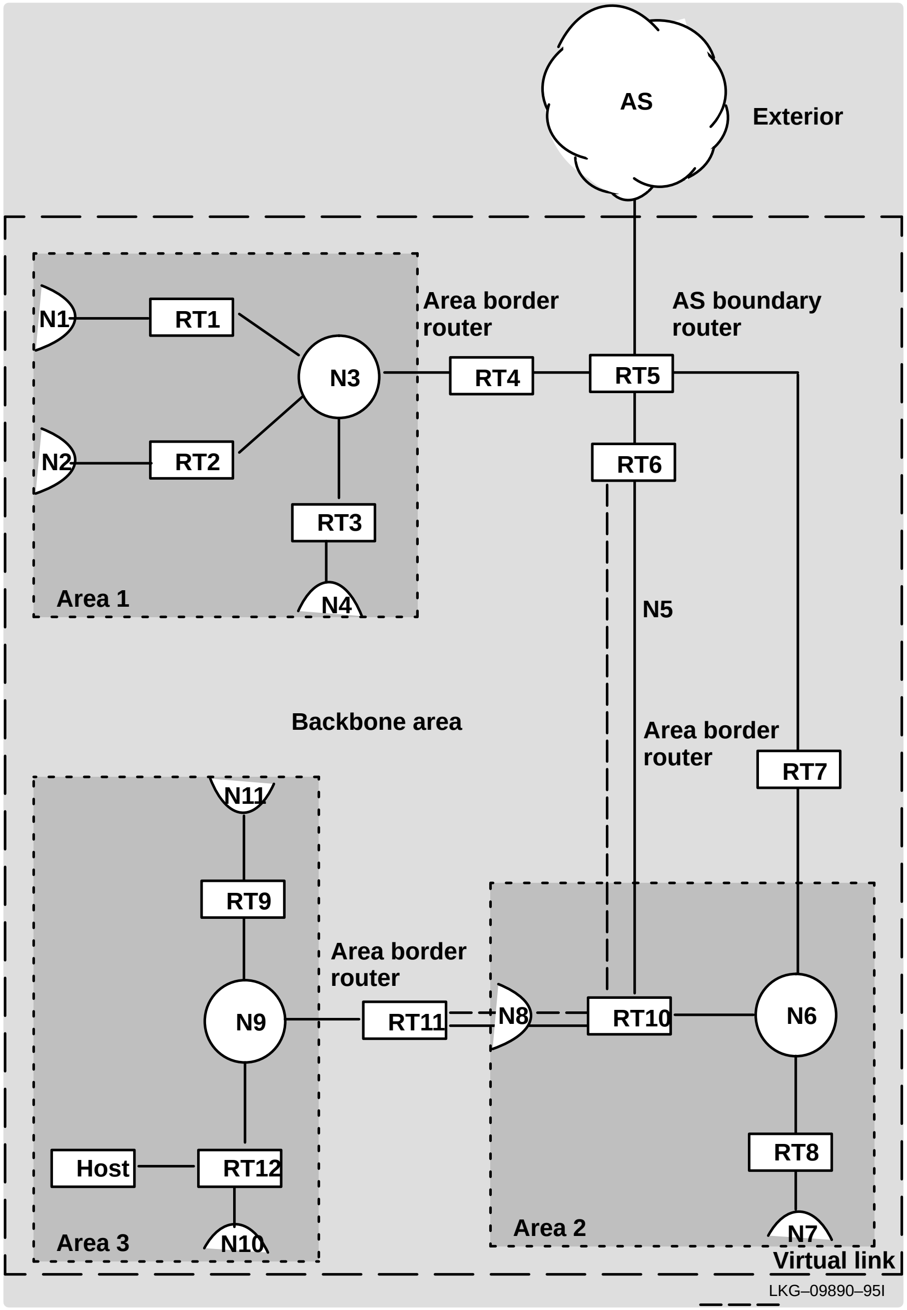
OSPF areas are defined as address ranges. External to the area, a single route is advertised for each address range. For example, if an OSPF area consisted of all subnets of the class B network 128.185.0.0, it consists of a single address range. The address range is specified as an address of 128.185.0.0 together with a mask of 255.255.0.0. Outside the area, the entire subnetted network is advertised as a single route to network 128.185.0.0.

OSPF Backbone Area

Every OSPF routing domain must have a backbone. The backbone is a special OSPF area having an area ID equal to 0.0.0.0.   The OSPF backbone must be contiguous; however, it is possible to define areas where the backbone is not physically contiguous. When this situation exists, you must configure a virtual link to maintain the backbone's connectivity (Figure 2–7). You can configure virtual links between any two backbone routers that have an interface to a common non-backbone area.

Figure 2–7 illustrates how you can configure OSPF areas.

Figure 2–7 OSPF Areas



The backbone is responsible for distributing inter-area routing information. The backbone area consists of any of the following:

- Networks belonging to Area 0.0.0.0
- Routers attached to those networks
- Routers belonging to multiple areas
- Configured virtual links

Area Routing Hierarchy

OSPF routing in an AS occurs on the following three levels:

- Intra-area
- Inter-area
- Exterior

Intra-Area Routing

Intra-area routing occurs when a packet's source and destination address reside in the same area. For example, N1 and N2 in Area 1 of Figure 2–7. Information that is about other areas does not affect this type of routing.

Inter-Area Routing

Inter-area routing occurs when the packet's source and destination addresses reside in different areas of an AS, for example, N1 of Area 1 and N7 of Area 2 in Figure 2–7. OSPF does inter-area routing by dividing the path into three contiguous pieces: an intra-area path from source to an area border router; a backbone path between the source and destination areas; and another intra-area path to the destination. You can visualize this high-level of routing as a star topology with the backbone as hub and each of the areas as a spoke.

Exterior Routing

Exterior routes are paths to networks that lie outside the AS. These routes originate either from routing protocols, such as Exterior Gateway Protocol (EGP), or from static routes entered by the network administrator. The exterior routing information provided by EGP does not interfere with the internal routing information provided by the OSPF protocol.

AS boundary routers may import exterior routes into the OSPF routing domain. OSPF represents these routes as *AS external link advertisements.*

OSPF imports external routes in separate levels. The first level, called type 1 routes, is used when the external metric is comparable to the OSPF metric (for example, they might both use delay in milliseconds). The second level, called external type 2 routes, assumes that the external cost is greater than the cost of any internal OSPF (link-state) path.

Imported external routes are tagged with 32-bits of information. In a router, this 32-bit field indicates the AS number from where the route was received. This enables more intelligent EGP behavior when determining whether to re-advertise the external information to other ASs.

Types of OSPF Routers

There are two special kinds of OSPF routers, area border routers and AS boundary routers.

- **Area Border Routers** – A router attached to multiple areas that runs multiple copies of the basic algorithm, one copy for each attached area and an additional copy for the backbone. Area border routers condense the topology information of attached areas for distribution to the backbone. The backbone then distributes this information to other areas.

- **AS Boundary Routers** – A router that exchanges information with routers that belong to other ASs. These routers import this information to the OSPF routing domain in AS external link advertisements.

OSPF Routing Summary

When a router is initialized, it uses the hello protocol to send hello packets to its neighbors, and they in turn send their packets to the router. On broadcast and point-to-point networks, the router dynamically detects its neighboring routers by sending the hello packets to the multicast address ALLSPFRouters; on non-broadcast networks you must configure information to help the router discover its neighbors. On all multi-access networks (broadcast and non-broadcast), the hello protocol also elects a designated router for the network.

The router then attempts to form adjacencies with its neighbors to synchronize their topological databases. Adjacencies control the distribution (sending and receiving) of the routing protocol packets as well as the distribution of the topological database updates. On a multi-access network, the designated router determines which routers become adjacent.

A router periodically advertises its status or link state to its adjacencies. Link state advertisements flood throughout an area ensuring that all routers have exactly the same topological database. This database is a collection of the link state advertisements received from each router belonging to an area. From the information in this database, each router can calculate a shortest path tree with itself designated as the root. Then the shortest path tree is used to generate the routing table.

Hello Protocol

A hello protocol periodically sends a hello packet out all router interfaces to establish and maintain neighbor relationships, as well as bidirectional communication. Bi-directional communication is ensured when a router receives a neighbor’s hello packet and sees itself on a listing. On a multi-access network, the hello protocol elects a designated router that has the job of determining what adjacencies form over the network.

The hello protocol works differently on broadcast networks and non-broadcast networks. On a broadcast network, each router discovers its neighbors dynamically by advertising itself by periodically multicasting hello packets. These hello packets contain the router’s view of the designated router’s identity and a current list of active routers.

On a non-broadcast network, the hello protocol needs to have network information statically configured to ensure the correct operation of the hello packet. With this type of network, any router that may potentially become a designated router has a list of all other routers attached to the network. A router with a designated router’s potential sends hello packets to all other potential designated routers in an attempt to find the designated router for the network. If the router itself is elected as designated router, it then sends hello packets to all other routers attached to the network.

Once a neighbor is discovered and bidirectional communication is ensured (on a multi-access network a designated router must also be elected), a decision is made as to whether to form an adjacency with the neighbor. An attempt is always made to establish an adjacency over point-to-point networks and virtual links.

Designated Router

Every multi-access network has a designated router that performs two main functions for the routing protocol, it originates network link advertisements and it becomes adjacent to all other routers on the network.

When a designated router originates network link advertisements, it lists all the routers, including itself, currently attached to the network. The link ID for this advertisement is the IP interface address of the designated router. By using the network/subnet mask, the designated router obtains the IP network number.

The designated router becomes adjacent to all other routers and is tasked with synchronizing the link state databases.

The hello protocol elects the designated router after determining the routers priority from the Rtr Pri field of the hello packet. When a router's interface first becomes functional, it checks to see if the network currently has designated router. If it does, it accepts that designated router regardless of that router's priority, otherwise, it declares itself the designated router. If the router declares itself the designated router at the same time another router does, the router with highest router priority (Rtr Pri) becomes the designated router. In the case that both Rtr Pris are equal, the one with the higher router ID is elected.

Once the designated router is elected, it becomes the endpoint for many adjacencies. On a broadcast network this optimizes the flooding procedure by allowing the designated route to multicast its Link State Update packets to the address ALLSPFRouters rather than sending separate packets over each adjacency.

OSPF and IP Multicast Routing

Multicasting is a LAN technique that allows copies of a single packet to pass to a selected subset of all possible destinations. Some hardware, for example, Ethernet, supports multicast by allowing a network interface to belong to one or more multicast groups.

The IP protocol supports IP multicast routing through IP multicast extensions to OSPF (MOSPF). IP multicast is an extension of LAN multicasting to a TCP/IP Internet. This process lets an IP host send a single datagram (called an IP multicast datagram) that is delivered to multiple destinations. IP multicast datagrams are those packets whose destinations are Class D IP addresses. Each Class D address defines a multicast group.

The Internet Group Management Protocol (IGMP) is the OSPF extension that lets an IP host participate in IP multicasting. IGMP lets routers keep track of IP group membership on its local LANs by sending IGMP Host Membership Queries and receiving IGMP Host Membership Reports.

An MOSPF router then distributes group location information throughout the routing domain by flooding a new type (type 6) of link state advertisement, the group-membership-LSA. This enables the MOSPF routers to efficiently forward a multicast datagram to its multiple destinations. This is done by each router calculating the path of the multicast datagram as a *tree* whose root is the datagram source and whose terminal branches are LANs containing group members.

While running MOSPF, multicast datagram forwarding works in the following ways:

- While forwarding IP multicasts is not reliable, IP multicast datagrams are delivered with the same *best effort* as with the delivery of IP unicasts.

- Multicast datagrams travel the shortest path between the datagram source and any particular destination (OSPF link state cost). This occurs because a separate tree is built for each datagram source and destination group pair.

- A multicast datagram is forwarded as a data-link multicast at each *hop*. The ARP protocol is not used. For some network technologies, mapping between Class D addresses and data-link multicast occurs while for others, Class D addresses are mapped to the data-link broadcast address.

- When paths from the datagram source to two separate group members share an initial common segment, only a single datagram is forwarded until the paths go in separate directions. The path can split at either a router or at a network. If the path splits at a router, the router replicates the packet before it is sent. If the path splits at a network, it replicates through a data-link multicast.

- You use MOSPF routers with OSPF routers that do not support multicast extensions. In this configuration, all routers inter-operate in the routing of unicasts. This allows you to slowly introduce multicast capability into an internetwork.

Note: Some configurations of MOSPF and non-MOSPF routers may produce unexpected failures in multicast routing.

- You construct separate multicast paths in MOSPF for each TOS. However, routers do not support TOS-based routers. You can mix non-TOS routers with TOS-based routers but this causes TOS to be ignored in the forwarding of multicasts.

- You configure the router to send SNMP traps to a multicast group address by adding a group address to a particular SNMP community name.

The RIP Routing Protocol

RIP is a distance-vector protocol (based on the Bellman-Ford technology) that allows routers to exchange information about destinations for computing routes throughout the network. Destinations may be networks or a special destination used to convey a default route.

Bellman-Ford algorithms make each router periodically broadcast its routing tables to all its neighbors. Then a router knowing its neighbors' tables can decide to which destination neighbor to forward a packet.

RIP Limitations

RIP is primarily intended for use in homogeneous networks of moderate size. Because of this, RIP has the following limitations:

- AS diameter limitation of 15 hops.

- RIP's metric (hop count) cannot adequately describe variations in a path's characteristics, sometimes resulting in sub-optimal routing.

- Slow to find new routes when the network changes. This search consumes considerable bandwidth, and in extreme cases exhibits a slow convergence behavior referred to as a *Count to infinity.*

Note: All bridging router interfaces running RIP must have the same subnet mask.

Customizing RIP

Occasionally it is necessary to customize RIP behavior. In IP, you can customize RIP with a number of configurable flags. Most flags take effect on a specified IP interface address. These flags control sending and receiving RIP information about each router interface.

The set of routes sent out from a particular address is the union of the routes selected by setting any of the following four flags. Subnet-level routes are sent only when the destination subnet is a member of the same IP network as the sending address.

Send Net Routes If set, the router sends all network-level routes.

Send Subnet Routes If set, the router sends appropriate subnet-level routes.

Send Default Routes If set, the router advertises a default route if the router itself has a default router in operation. The route to the default (authoritative) router is advertised as a route to destination 0.0.0.0.

Send Static and Direct Routes If set, the router advertises all directly connected networks and statically configured routes.

The following five flags control how information received by RIP is incorporated into the router’s routing tables. Certain flag settings allow RIP routes to override static routing information, but only if the RIP metric is better than the static route’s metric.

Override Default If set, RIP packets received on this IP interface may override the router’s default gateway.

Override Static Routes If set, RIP packets received on this IP interface may override any of the router’s statically configured routing information.

RIP Input Off If set, RIP packets received on this IP interface are ignored.

Receive Dynamic Net Routes When not set, the router accepts RIP updates only for those networks that are specified in an `IP config>` **add accept-rip-route** configuration command.

Converting RIP to OSPF

To convert your AS from RIP to OSPF, leave RIP running and install OSPF separately on each router. Gradually, all your internal routes shift from being learned by RIP to being learned by OSPF (OSPF routes have precedence over RIP routes). To ensure that your routes look exactly as they did before the conversion, set the cost of each OSPF interface to 1. This ensures that the hop count is used as your OSPF metric.

When installing OSPF, remember to estimate the size of your OSPF system when the protocol is enabled. This size estimate reflects the final size of the OSPF routing domain. When the installation is complete, turn on AS boundary routing in all routers that still need to learn routes through other protocols (EGP, RIP, and statically configured routes). Keep the number of AS boundary routers to a minimum.

Exterior Gateway Routing Protocol

The Exterior Gateway Protocol (EGP) exchanges routing information with other routers in different ASs. To accomplish this, the router establishes sessions with a number of peers, called EGP neighbors. EGP uses periodic polling containing Hello/I-Heard-You (I-H-U) message exchanges to monitor neighbor reachability and to poll commands to solicit update responses. EGP is based on a finite state machine model with defined events, state transitions, and actions.

EGP States

There are five states that establish the state of the protocol between the router and each of its neighbors: idle, acquisition, down, up, and cease. A router maintaining an EGP connection with a number of neighbors must maintain a separate set of states for each neighbor. The current state, events, and actions of the state machine apply to each neighbor separately.

Determining Neighbor Reachability

The purpose of the neighbor-reachability algorithm is to confirm that the neighbor is operational and capable of providing reliable net-reachability information. An equally important purpose is to filter noisy reachability information before sending it on to the remainder of the AS, therefore avoiding unnecessary reachability changes.

The neighbor-reachability algorithm runs continuously whether the gateway is in the down or the up state. This algorithm is used in either the active or passive mode. In the active mode the router sends periodic hello commands and listens for I-H-U responses in order to determine neighbor-reachability. In the passive mode hello commands are not sent and I-H-U responses are not expected. Reachability is indicated in the *status* field of received hello commands. Poll commands and Update responses are used in place of hello commands and I-H-U responses respectively, since they contain the same *status* field information.

IGP-EGP Routing Information Interchange

An IGP-EGP routing interchange allows routers from different ASs to know the reachability of other network segments in that AS.

When your router is using both an IGP (OSPF, Integrated IS-IS, or RIP) and EGP, you can determine how routing information flows between the IGP and EGP. This flow tells you how widely routing information is propagated through the network.

The interchange of routes is defined for each AS; so, if the router is speaking EGP to several peers (EGP neighbors) each belonging to a different AS, a separate interchange is defined for each neighbor. Both directions of the interchange are table driven. The list of routes that EGP advertises is specified by the Output Exchange Table, configurable through the router’s user interface.

Similarly, the set of EGP-learned routes to be advertised through the IGP is specified by the Input Exchange Table. This table is also configurable.

Both the Input and Output tables allow you to specify the metric to be advertised on a route-by-route basis.

You can also allow the free exchange of routes in any one direction by setting the IGP-EGP interchange flag to either *in* or *out*. For example, if the IGP-EGP interchange flag is set to *in*, all EGP learned routes are advertised through the IGP, and the Output Exchange Table specifies which routes are advertised through EGP. You can configure the IGP-EGP exchange in more flexible ways. For instance, you can specify that all routes learned from a particular AS be advertised to another AS; or you can specify that a route be advertised to a particular AS, but only if the route was received from some other AS.

3

The SNMP Protocol

This chapter describes the Simple Network Management Protocol (SNMP) protocol.

SNMP Overview

SNMP is an OSI layer 7 (application layer) protocol for monitoring router operating characteristics. Use SNMP with software supplied by the user running on a remote host.

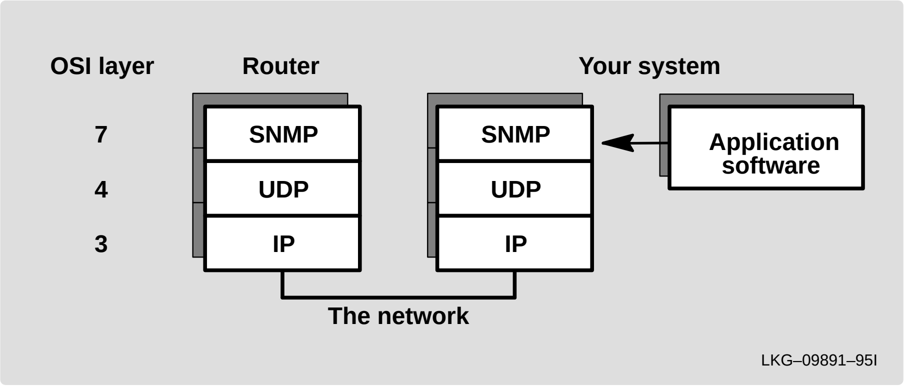
SNMP enables network hosts to read and modify some of the settings of the router's operating characteristics. It allows software running on a remote host to contact the router over a network and get up-to-date information about the router on request. Since SNMP software can access most of the configuration data, you do not have to type in commands at a remote console.

SNMP's basic functions include the following:

- Collecting information and modifying router operating characteristics on behalf of remote SNMP users.
- Sending and receiving SNMP packets through the IP protocol.

Figure 3–1 shows the SNMP protocol environment.

Figure 3–1 Protocol Layers of the SNMP Environment



The software that processes SNMP requests runs on the router. The user program that makes SNMP requests runs on the user's machine elsewhere in the network, not on the router. The SNMP agent at the router and the user program both use the UDP/IP protocol to exchange packets.

For more information about SNMP, refer to RFC 1157, *A Simple Network Management Protocol*. Refer to RFCs 1212 and 1213 for descriptions of SNMP variables. The RFCs explain how to use the protocols and formats of the packets that the protocols employ. RFCs are available from the Network Information Center (NIC) at Government Systems, Inc., Chantilly, Virginia.

SNMP Packet Types

SNMP's packet types reflect SNMP's basic functions. The packet types include the following:

- **GET REQUEST packet** – Travels user-to-router. Contains requests by user software for information. Retrieves the exact variable requested.
- **GET NEXT REQUEST packet** – Travels user-to-router. Contains information requests by user software. Retrieves the next alphabetically higher variable.
- **SET REQUEST packet** – Travels user-to-router. Contains requests by user software to modify router operating characteristics.

- **GET RESPONSE packet** – Travels router-to-user. Contains the target router’s response to a GET NEXT REQUEST or SET REQUEST packet, sent by the user software.

- **TRAP MESSAGE packet** – Travels router-to-user. Contains unsolicited information from the router and numerically encoded messages, such as `An interface on the router went up/down` or `The router reloaded its software`.

Authentication

Authentication prevents unauthorized users from learning information about a router or modifying its operating characteristics. In particular, the authentication protocol ensures that both the router SNMP server and the remote SNMP application ignore and discard requests from unauthorized users.

To determine if an incoming message represents a legitimate request by an authorized user or an accidental or malicious request by an unauthorized user, SNMP uses various sets of rules. Each set of rules is called an *authentication scheme*. Authentication schemes rely on mathematical or cryptographic techniques to authenticate messages.

For each SNMP community, you select an authentication scheme for users of that session. The current implementation of SNMP offers a single authentication scheme called *trivial.*

For more information about creating and using authentication schemes with SNMP, refer to RFC 1057, *A Simple Network Management Protocol*.

4

The IPX Protocol

This chapter describes router implementation of the IPX protocol.

IPX Overview

IPX is a collection of software components that transfers information between networks connected by physical media. By transferring information, IPX software facilitates communication between network devices, such as personal computers, file servers, and printers.

This implementation of IPX allows the router to function as a Novell NetWare internetwork router. It is functionally compatible with the bridging function in a NetWare file server and has a standalone NetWare bridge.

IPX Addressing

An IPX address specifies the location of a particular entity in a network or internetwork. Addresses allow two entities that are not directly connected to communicate. Each entity, such as a host, server, communication device, or printer in a network or internet, must have a unique identifier, or address.

A simple solution is to use multi-part addresses, like the city-street-house address on a piece of mail. For example, the IPX protocol refers to network numbers (city), node numbers (street), and socket numbers (house).

- **Host Number** – The unique hardware address required by each node on the network. An example of a hardware address is a 48-bit Ethernet node address. Proper node addressing ensures that the network efficiently delivers and receives packets. Each IPX interface uses its right-justified hardware node address as its 48-bit host number. The hexadecimal address FF-FF-FF-FF-FF-FF is the broadcast address.

- **Network Number** – The unique address required by each IPX network. The network number is a 32-bit hexadecimal number. In IPX, this number is also called the *network address*. You must configure the router with the IPX network number for each IPX network interface. Use the same network number for all routers and file servers on a network, but use different numbers for each network.

- **Socket Number** – The location within the protocol that binds the packet to an application service.

In general, IPX networks can be separated into two physical elements: Local and Remote networks.

IPX Routing

A *route* indicates the path an entity's packet follows to reach another entity. IPX uses the RIP protocol to maintain the routes in its routing tables. Valid entries remain in the routing tables for three multiples of the RIP interval. (The default value of 60 seconds allows a valid entry to remain in the table for 180 seconds.) During this time, if a route entry is not refreshed by RIP updates, the route is marked with an infinite hop count (16).

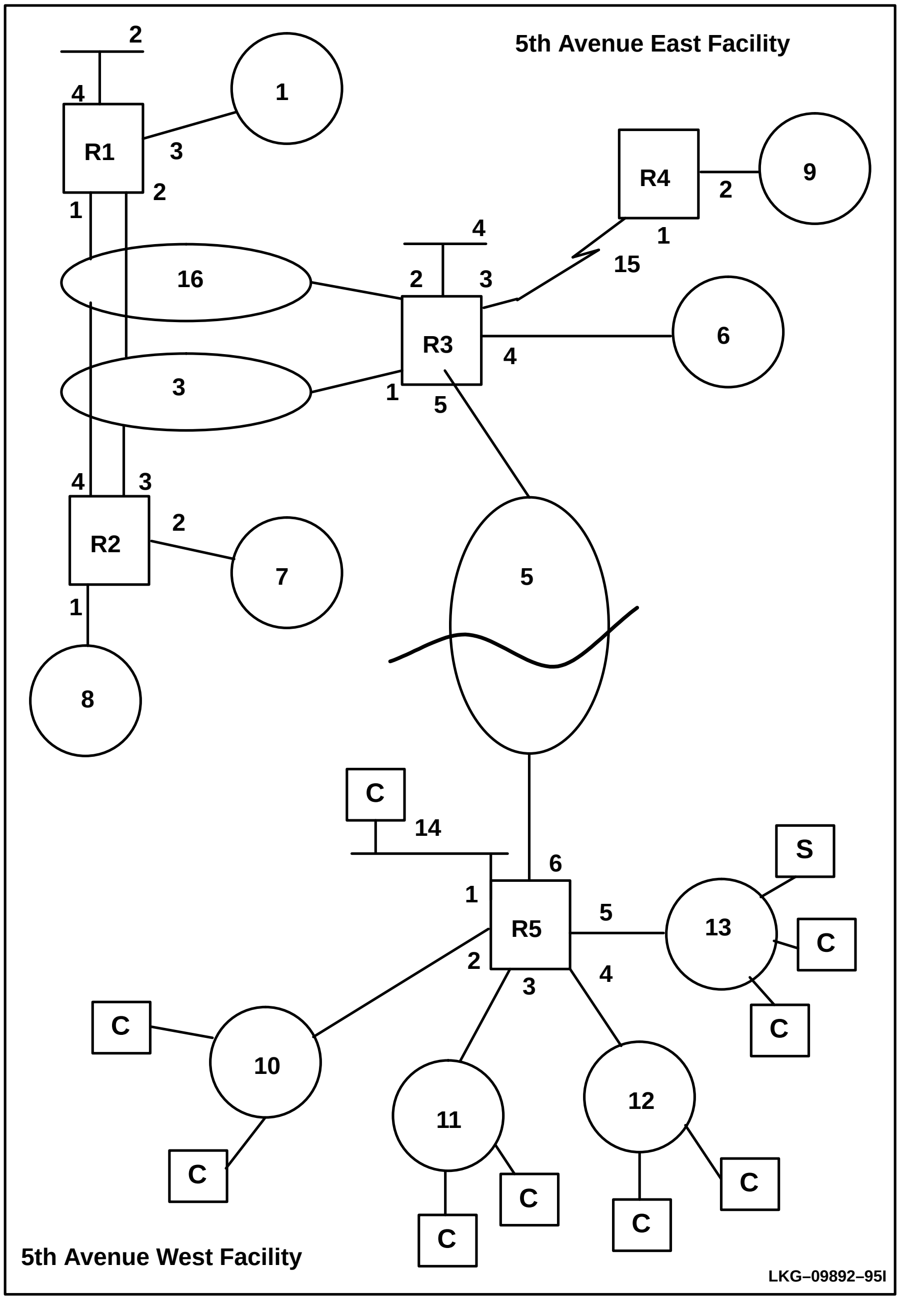
Any packet locally addressed to the router, including broadcast packets, is passed to the appropriate internal module for processing. Examples of this are SAP packets and RIP packets. Broadcast messages that are sent to an un-implemented socket do not elicit error replies.

If the packet size is greater than the output size of the next hop network, the router discards the packet and returns an error message.

Local vs Remote Networks

Local networks are networks to which the router is directly attached. Remote networks are networks that are from 1 to sixteen router hops away from the router being considered. In Figure 4–1, Router 5 has six interfaces. The IPX networks attached to these interfaces are local and are termed “client” networks. The router addresses each client on these networks at the MAC layer, as well as at the IPX network layer. IPX network 6 is a remote network that Router 5 accesses through its interface 6.

Figure 4–1 Sample Network Map



Alternate Routes

In Figure 4–1, Router 5 has a single connection to the rest of the network (a token ring with IPX network number 5). Therefore, alternate routes to other networks such as IPX network 6 (through Route 3) do not exist. Set the maximum alternate-routes-per-destination to 1. In addition, there are ten remote IPX networks that can be reached. Consequently, the maximum networks default of 32 is more than adequate. The Maximum total-alternate-route-entries for this router is the number of alternate routes needed to be kept (0). The minimum is 1.

Router 1 has two possible paths it uses to reach the other two routers on the redundant backbones (for example, token rings with IPX network numbers 3 and 16). Therefore, it has an alternate route to the remote networks beyond the other two backbone routers. Configure Router 1 to:

1. Support at least 1 alternate-route-per-destination remote network.
2. Set the minimum total-alternate-route-entries to 12. (IPX networks 4 through 15 each need to have storage for their alternate routes.)

Note: If there are 3 token-ring backbones, an additional 12 alternate routes exist. This requires that total-alternate-routes-entries be set to 24 and the alternate-routes-per-destination be set to 2.

IPX RIP Interval

The IPX Routing Information Protocol (RIP) interval permits users to configure the interval between RIP updates on any interface. The interval can be varied from the original Novell default of 60 seconds to 24 hours. This allows users to reduce traffic on heavily used low speed WAN lines and dial circuits. All router interfaces on the same IPX network must use the same RIP interval.

Note: While complete RIP advertisements are controlled by the interval, network topology changes are still propagated across the interfaces as quickly as they are learned.

Note: The RIP interval is not configurable on the Novell file server.

Multiple Routes

The current version of the RIP protocol supports multiple routes to a given destination instead of only one. This feature provides a more stable IPX configuration.

Previously, if multiple best routes existed to a target network, RIP kept only one. If the route is lost, the router sends out unreachable network RIP broadcasts. This condition then ripples down the network. When the next RIP packet arrives from the alternate route, the router learns the new route and sends another RIP packet announcing that to the world. The current RIP deletes only a single table entry. The other route automatically becomes the route of choice.

IPX Tunneling

The IP tunnel is a pseudo network interface that is similar to a real network interface without the MAC layer. The lowest layers of the IP tunnel network interface are made up of UDP and IP protocol functions. The IP tunnel interface is instantiated after all conventional interfaces are allocated. The IP tunnel interface is dynamic and its interface number can change as more interfaces are configured after the IP tunnel.

The IPX forwarder is configured on the IP tunnel network interface in the same way as an ethernet or token ring interface. The interface is assigned an IPX network number with associated protocol timers. The router’s internal IP address replaces the IPX host identifier (the LAN MAC address). A list of peer IP unicast addresses can be used to join multiple IPX routers on the IP tunnel pseudo network. The source host identifier and the list of peer addresses are the basis for the tunnel addressing mechanism.

Because the IPX peer IP address list is static, it must be configured the same on all IPX routers that use tunneling. Different IP address lists on the same IP pseudo network can cause inconsistent forwarding. While you can configure multiple peer groups on the same pseudo network, you cannot have multiple peer groups on the same IPX router.

You can add a single IP multicast address that represents the IPX router group peer list. The tunneling mechanism registers the IP multicast class D address with participating MOSPF routers using group membership protocol, IGMP. You can choose any class D address between 224.0.0.2 and 239.255.255.255 for tunnel group membership.

If you configure an IP multicast group address for the tunnel, you invalidate any IP unicast peer addresses that were previously configured.

Service Advertising Protocol (SAP)

The IPX router follows the Service Advertising Protocol (SAP). SAP is a distributed database used to find NetWare Services such as file servers. Services are uniquely identified by a 2-byte numeric type and a name of up to 48 characters. Each service provider advertises its services, such as type, name, and address. The router accumulates this information and sends it to other routers.

IPX SAP Interval

The IPX Service Advertising Protocol (SAP) interval permits users to configure the interval between SAP updates on any interface. The interval can be varied from the original Novell default of 60 seconds to 24 hours. This allows users to reduce traffic on heavily used low speed WAN lines and dial circuits. All router interfaces on a given network must use the same SAP interval.

Note: While complete SAP advertisements are controlled by the interval, network services changes are still propagated across the interfaces as quickly as they are learned.

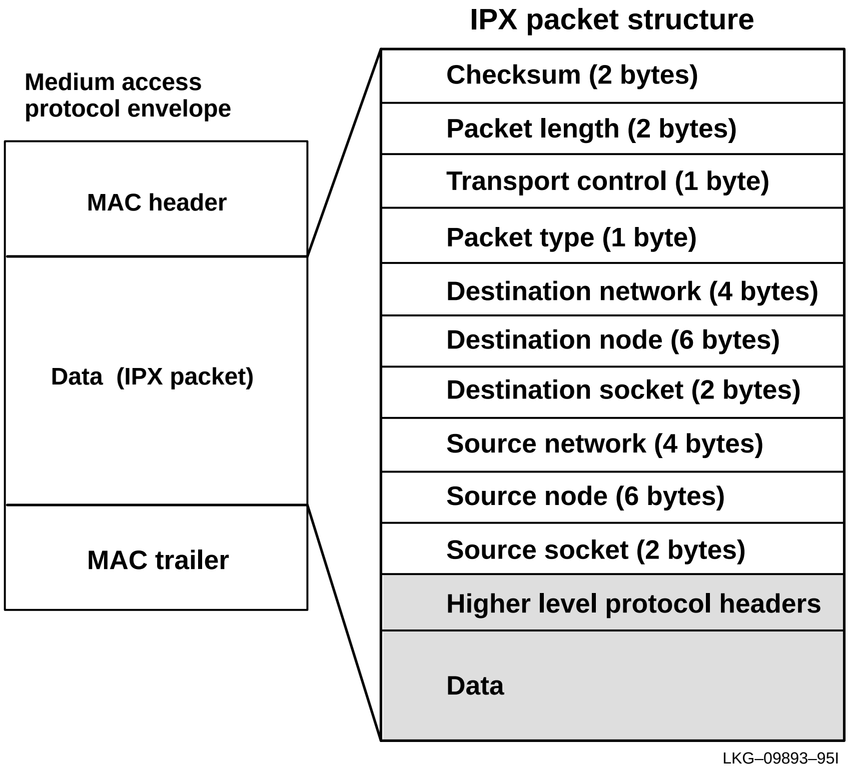
Note: The SAP interval is not configurable on the Novell file server.

IPX Encapsulation

As an IPX packet travels across a media interface, it is encapsulated within an envelope specific to the type of medium on which the packet is transmitted. For instance, an IPX packet traveling over a Token-ring network does not look like an IPX packet traveling over an Ethernet network.

In Figure 4–2 below, the IPX packet is enveloped by a Media Access Control (MAC) header and Data Link Header. The MAC header, and all of the trailer, are specific to the particular medium being used. The Data Link Header varies by MAC type.

Figure 4–2 IPX Packet Structure



IPX Encapsulation Types

Novell has defined the following set of encapsulations that can be used with IPX:

- **Token ring**

  IEEE 802.2 SAP – The current default token ring encapsulation.
  IEEE 802.2 SNAP – Used primarily for bridging environments.

  Token ring also has the option of specifying the use of canonical (LSB) or non-canonical (MSB) bit order in its addressing. The default bit order for Digital routers and Novell NetWare is MSB non-canonical. The use of LSB bit order allows bridging IPX between token ring and other network types.

- **FDDI**

  IEEE802.2 SAP
  IEEE802.2 SNAP – The current default for Digital routers.

- **Ethernet**

  IEEE 802.2 SAP – Default for NetWare 4.0 and later.
  IEEE 802.2 SNAP – Use of this is not recommended. It violates IEEE 802.1 (Bridging Ethernet in IEEE 802 environment). There is no route caching performed on this encapsulation.

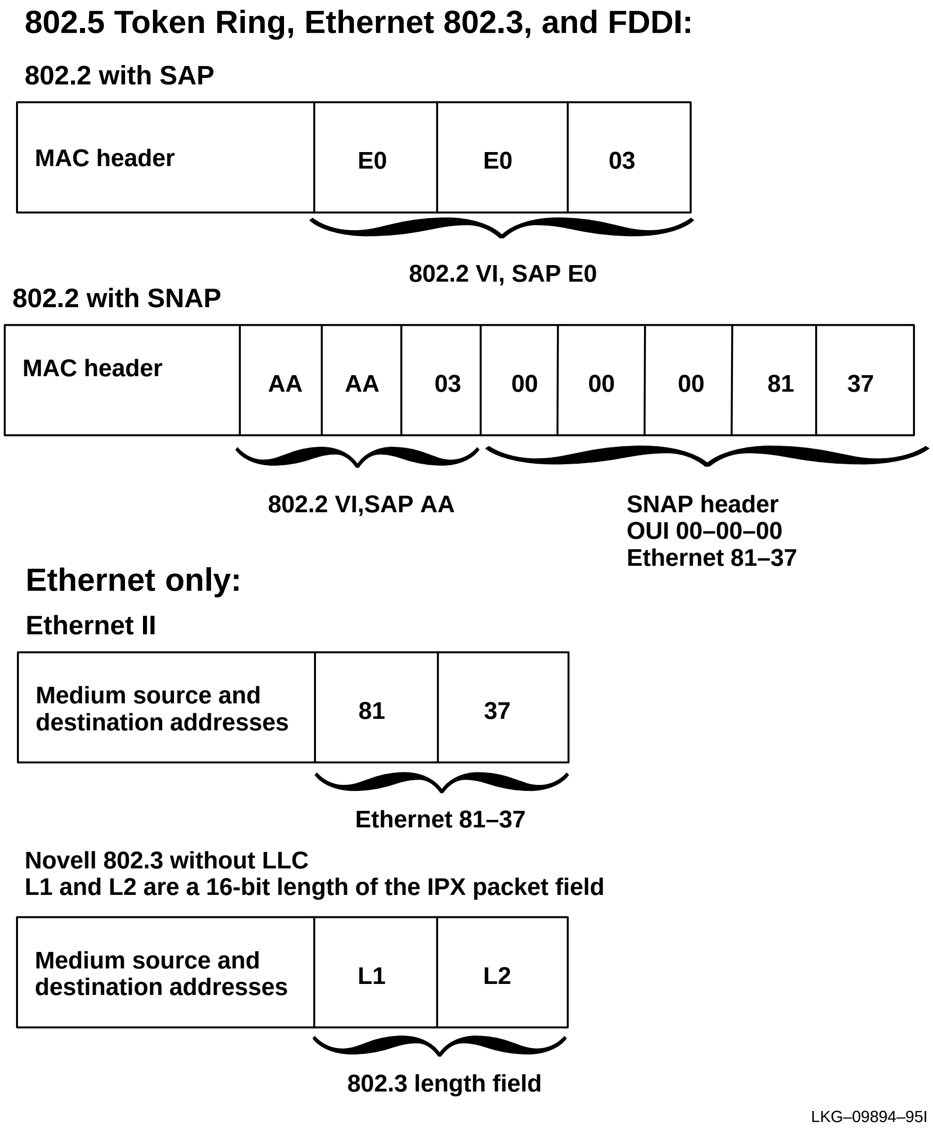
  Ethernet II – Old NetWare Ethernet encapsulation.
  Novell (802.3 without LLC) – Default for NetWare prior to version 4.0 (not bridgeable).

  If you have NetWare workstations or servers, or both, and are using pre-ODI V1.12 Ethernet network drivers, the only encapsulations that can be used are Ethernet_II, Novell (802.3 without LLC), and token ring_swap MSB.

Note: The Ethernet_8023 encapsulation cannot be used on any network in an IPX internet where IPX end-to-end check sums are in use.

Figure 4–3 shows the different encapsulations.

Figure 4–3 IPX Encapsulation



Encapsulation Usage

The encapsulation for IPX must be the same for all nodes on an IPX network that you want to interoperate. You can create more than one IPX network on a given physical network (token-ring or Ethernet segments), logically separating them by assigning different encapsulations to the members of each virtual network. Generally, IEEE 802.2 SAP is the standard encapsulation to use in routing environments.

Note: With bridges, the IEEE 802.2 SNAP encapsulation is recommended. Uniformly using a SNAP encapsulation on bridged networks simplifies the conversion logic within the bridge. SNAP's can be copied, whereas the bridge must be told how to convert unusual encapsulations, such as Novell 802.3 with or without LLC, as they are bridged onto a different medium.

NETBIOS

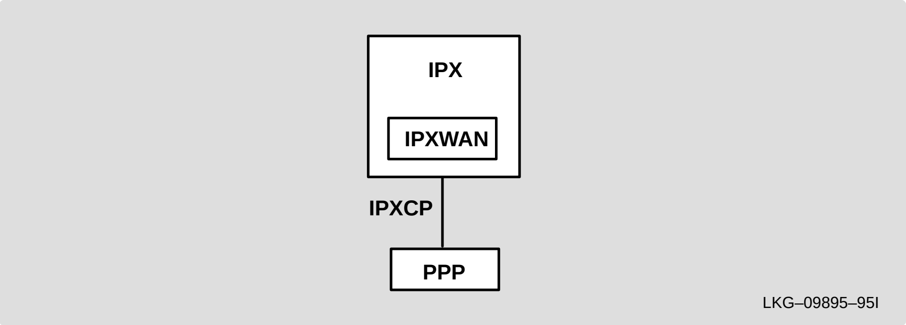
The Novell NetBIOS emulator requires special support from bridges and routers. Novell bridges, routers, and the IPX protocol provide this support as of Version 2.12.

This support allows the NetBIOS naming functions to operate across routers. There is a limit of eight networks between two nodes that communicate using NetBIOS.

IPXWAN: IPX Over the Point-to-Point Protocol

IPXWAN allows you to exchange configuration information from router to router over WAN networks. This exchange of information occurs prior to exchanging standard Internetwork Packet Exchange (IPX) routing information and traffic over the WAN. IPXWAN is part of IPX. IPX currently supports IPXWAN using IPXCP (IPX Control Protocol) over the Point-to-Point Protocol (PPP) as shown in Figure 4–4.

Figure 4–4 IPXWAN over Point-to-Point Protocol



IPXWAN requires a router to have a primary network number that is unique to the entire internet, and permanently assigned to that router.

Each IPXWAN router must have a router name (file server name). This symbolic name given to the router must be from 1 to 47 characters long and can contain the characters A through Z, underscore(_), hyphen (–), and "at" sign (@).

IPXWAN operation includes the following tasks:

- Negotiating master and slave roles for IPXWAN exchange. (IPXWAN is a master/slave protocol where the master is the requester in the exchange of request/response packets.)
- Information exchange of final router configuration. (IPXWAN allows for the selection of which protocol to use, however Digital routers only support RIP/SAP.)

The **ipxwan** console command allows you to list the current configuration information or a summary of configuration information for IPX running over a WAN interface using the point-to-point protocol.

For further information on how IPX operates over various WAN media including PPP refer to the *Request For Comments: RFC 1362* available from IETF.

IPX Access Controls

IPX access controls can be used to prevent the router from forwarding packets based on IPX addresses (net/host/socket). This can be used for security, to protect valuable services. It also can be used to stop the forwarding of packets from "noisy" applications beyond the area of interest.

The access controls are based on the originating IPX source address, and the ultimate destination IPX address. Intermediate hop addresses are unimportant.

An IPX address (source or destination) address for access controls consists of an IPX network number, an IPX host number, and a range of IPX socket numbers. All are specified in hexadecimal. The network number and host number can be specified as 0, which is a wild card that matches all network and host numbers. A range of 0 to FFFF is a wild card for sockets.

The access control list is an ordered list of entries. Each access control entry can be inclusive or exclusive. An inclusive entry, if it matches the packet, allows it to be forwarded, and terminates the search of the list. An exclusive entry, if it matches the packet, causes it to be dropped, and terminates the search. If the end of the list is reached, the packet is dropped. (This is equivalent to having a wild-card exclusive entry at the end of the list.)

When devising access control lists, certain things about the IPX protocol must be considered. First, never block the RIP and SAP sockets (0x0453 and 0x0452). Blocking these completely breaks the operation of the IPX protocol.

Also, recognize that the access control list is global, and applies to all interfaces. Use source or destination network numbers in the access controls to enact directional controls.

It is also important to understand where the services you are trying to protect are located. Most services are advertised through the SAP protocol, so you can use that information to locate the address (net/host/socket) of the service you are protecting (or allowing). Use the output of the routers `IPX>`**slist** command to determine the address of a service.

Note: All services on a Novell file server (version 3.0 or higher) are on the server’s internal network, usually at host 000000000001 on that network. Since that internal network number is unique over an entire IPX network, you can protect it by blocking all packets to the internal network, host 000000000001, socket range 0–FFFF. To only block the file server, use a socket range 0451–0451.

When extracting socket numbers from an **slist** to build an access control list, remember that some services have fixed socket numbers, and some have dynamic (temporary) socket numbers. Sockets in the range 4000–7FFF are dynamic, there is no guarantee that the service has the same socket number the next time the file server is rebooted. However, socket numbers in the range 8000–FFFF are assigned by Novell, and generally remain constant.

IPX SAP Filtering

IPX SAP filters are a way of preventing service advertising information from being propagated through the router. There are three primary reasons to use SAP filters:

- If you are using NetWare Version 2.15 or lower servers, with small bindery sizes, and must limit the amount of information in the SAP database.
- If you do not want to advertise certain services outside the local area, since remote access to them is inappropriate.
- If you want to remove clutter from the SAP table.

Note: None of these reasons explicitly mention security. SAP filters cannot protect a service. All that the SAP protocol does is provide a name to address translation for services. If a potential intruder knows the address of the service, blocking its advertisement through SAP does not protect the service. Only access controls can provide security.

The SAP filter is based on setting a maximum hop count for a particular service, or group of services. Any matching service advertisement received with the specified hop count (or less) is accepted into the SAP table. Others are ignored. Only those services in the SAP database are readvertised to other services, or used to answer queries.

A SAP filter can apply to all services of a type. Novell assigns four digit hexadecimal type numbers for each type of service.

Alternatively, a SAP filter can apply to one particular service of a type. This is done by specifying the name of the service. (Note that the router only allows service names to be entered in 7-bit ASCII. Some service names use binary data, in violation of Novell SAP specifications. You cannot filter these services by name.)

A given SAP type can have several filters for a specific name, or one (wild card) filter for all names.

To determine the SAP type for a filter you want to establish, use the `IPX>`**slist** command. If you know the name of the service you want to filter, precede it with the type. Table 4–1 lists some common SAP types.

Table 4–1 SAP Types

SAP Type	Name
0004	File Server
0021	NAS SNA Gateway
0023	NACS Async Gateway
0027	TCP/IP Gateway
0029	Eicon 3270 Gateway
0047	Advertising Print Server
004B	Btrieve VAP V5.0
0050	Btrieve VAP V4.11
007A	TES–Netware for VMS
009A	Named Pipes Server
009E	Portable NetWare
0107	Remote Console – RCONOSLE NLM
0304	Novell SAA Gateway
039B	Lotus Notes

The list shown in Table 4–1 can vary. To verify SAP filter changes, follow these steps:

1. At the `IPX>` prompt enter the **slist** command. Note the entry for the service(s) you want to no longer propagate.

2. Add a SAP filter for that service type (optionally name), and the appropriate hop count.

3. Restart the router.

4. Check that the service is no longer listed in an `IPX>`**slist**.

IPX Performance Tuning

The current version of IPX supports a “fast path” architecture to provide for more efficient routing of IPX traffic. The “fast path” routing concept is to split the forwarding path into two pieces.

The “fast path” is used only to forward data packets, while a slower path handles administrational packets which take longer to process. “Fast path” uses an address cache that contains pointers to low level information for a given net address. This enables a packet to be forwarded much more quickly.

The slower routing table look-ups are performed only during the installation of a cache entry, and are then reused for each packet sent to a given destination. The cache has an aging mechanism that allows overflows to be dealt with intelligently. The cache size is configured through the IPX configuration menu.

The IPX Fast Past Cache includes two entries: Local and Remote. Each one can handle the requirements of that type of addressing.

Under the “t 6” process, the cache commands are:

- Set local-cache size X (1<=X<=10000) (default = 64)

- Set remote-cache size X (1<=X<=10000) (default = 64)

The cache commands are used to set a limit on the maximum number of entries of a given type to allow in the cache.

The following describes local and remote cache based on the sample network map shown in Figure 4–1.

Local Cache. The size of the local cache is equal to the total number of clients on each router’s local or client networks. Using the example in Figure 4–1, router 5 has a total of 10 clients (9 clients plus the server). Based on this total:

1. Multiply that figure (we used 10 in our example) by 10%.

2. Add that total (1) to the client total (for a safety margin).
3. Use that new total (11) for the number of local cache entries.

When all of the cache entries are in use, the least frequently used entries are purged. Using a 10% buffer (calculated in Step 1), prevents excessive purge requests.

Remote Cache. The size of the remote cache is equal to the total number of remote networks used by the router. In Figure 4–1, there are 10 IPX networks that R5 can reach through IPX network 5. Therefore, R5 has a total of 10 remote networks. Based on this total:

1. Multiply that figure (we used 10 in our example) by 10%.
2. Add that total (1) to the remote network total (10) for a safety margin.
3. Use that new total (11) for the number of remote cache entries.

The cache entries can be seen using the IPX console **cache** command as described in the “Monitoring IPX” chapter.

5

The DNA Phase IV Protocol

This chapter describes the router implementation of Digital Network Architecture (DNA).

DNA Phase IV Overview

DNA Phase IV (DNA IV) is a collection of software components that transfer information between networks connected by physical media. By transferring information, DNA IV software facilitates communication between network devices, such as personal computers, file servers, and printers.

The DNA IV protocol is the underlying protocol for Digital's DECnet software products, as well as DNA-compatible products from other vendors. The DNA IV protocol includes the following information:

- Routing software for DNA IV protocol networks.
- NCP, an implementation of the DNA IV Network Control Program.
- Support for DNA IV Maintenance Operations Protocol (MOP).

DNA IV performs two major functions:

- It maintains a complete routing database on all nodes in its area. If the router is operating as a level 2 router, it maintains the database for all areas as well.
- It routes incoming DECnet data packets to the appropriate destinations based on its own routing database. It ignores packets that are addressed for the router that are not hello packets or routing packets.

DNA IV supports the following:

- Multiple areas on an Ethernet or token ring network.

- Basic MOP operations. DNA IV responds to a MOP Request ID message with a MOP System ID message. DNA IV also sends a MOP system ID Message when a circuit comes up. You can monitor MOP messages using the Ethernet configuration module under DECnet-VAX NCP. The router NCP does not include an Ethernet configuration module.

Give special consideration to the following DNA IV specifications:

- DNA IV does not support the NSP, Session, or NICE protocols.

- DNA IV does not support the DDCMP line protocol on its directly-connected synchronous lines.

- DNA IV does not provide any Phase III compatibility features since it does not support the DDCMP data link protocols used by all Phase III nodes.

- NCP (the router's implementation of the DECnet Network Control Program) implements a subset of the original NCP commands and functions.

NCP

The main user interface program for the router's implementation of DNA IV is called NCP. The router's NCP is a limited subset of the DECnet Network Control Program (NCP) commands. NCP allows you to view and modify the various operating arguments of DNA IV and to read various DNA-specific counters.

Some of the features of NCP include the following:

- NCP implements new entities: module access-control and module routing-filter.

- NCP has no **set executor buffer size** command since the router does not originate any DECnet traffic. The router can forward the largest packet any DECnet implementation can generate. It honors the buffer size restrictions of all adjacent nodes.

- NCP allows an **all** qualifier on the **node**, **area**, and **circuit** subcommands.

The router NCP is similar to the NCP on DECnet-VAX, with the following differences:

- NCP does not include the **set node name** command, and therefore cannot assign names to nodes, or display node names with addresses.

- NCP does not include the **clear** or **purge** commands, nor do the **set** commands have an **all** argument. The permanent database is always copied to the volatile database when the router starts, restarts, or boots.

- An NCP command can have only one argument.

- NCP does not have the concept of lines. To see the data that a DECnet-VAX NCP **show line** command displays, use the GWCON **interface** and **network** commands.

- NCP does not support cross-network commands:

  – NCP does not include the **tell** command, which requests NCP commands on other nodes.

  – Similarly, NCP does not support protocol requests from other DNA nodes to execute NCP commands at the router on their behalf.

DNA IV Terminology and Concepts

This section contains a brief discussion of the DNA IV terminology.

Addressing

Each node has a 16-bit node address, which is the same for all interfaces on that node. An address consists of 2 fields: 6 bits of area number and 10 bits of node number. Addresses are printed in decimal with a period separating the area and the node, such as 1.7 is node 7 in area 1. If no area is given, area 1 is assumed. Any address in the range 1.1 to 63.1023 is legal. Both nodes and areas are numbered starting from 1, with few, if any, gaps. This is because the maximum node number and the maximum area numbers are configuration options and control the size of many of the routing data structures.

There is no direct correlation between addresses and physical cabling. Routes are computed to nodes, not wires.

Ethernet Data Link Addressing

Each Ethernet interface is set to the same 48-bit physical address, which is the concatenation of a 32-bit prefix (AA-00-04-00) and the 16-bit DNA IV node address. The node address is byte-swapped. Thus, DNA IV node 1.1 has Ethernet Address AA-00-04-00-01-04.

Multicast (not broadcast) is also used in routing. The three multicast addresses used by DNA IV are AB-00-00-02-00-00, AB-00-00-03-00-00, and AB-00-00-04-00-00.

802.5 Token Ring Data Link Addressing

The implementation of DNA over IEEE 802.5 Token Ring conforms to the *DECnet Digital Networking Architecture (Phase IV) Token-Ring Data Link and Node Product Functional Specification,* Version 1.0.0, that includes support for Arbitrary MAC Addresses (AMA).

Conventional DNA IV MAC addressing on token ring is the same as on Ethernet. Arbitrary MAC Addressing, on the other hand, allows the DNA protocol to run on IEEE 802.5 nodes without their MAC addresses being changed. This is necessary if you follow certain IBM protocol conventions. You can select the type of addressing with the NCP **define circuit** *cir_name* **router type** command.

The order in which bits are transmitted on token ring is the reverse of the way they are transmitted on Ethernet. Token ring MAC addresses are bitwise reversed within the same byte order. For example, the 32-bit prefix AA–00–04–00 is transmitted on token ring as 55:00:20:00. When MAC addresses are displayed, the bit order is indicated by the character used to separate the bytes. Dashes are used for Ethernet and colons are used for token ring.

X.25 Data Link Addressing

This implementation treats an X.25 network like an Ethernet. Most other DNA IV routers use DLM (data link mapping), which treats an X.25 virtual circuit as a point-to-point data link. A router can only communicate over X.25 with other nodes that use the same method (Ethernet-style or DLM).

Internal tables on each router connected to the X.25 network map the DNA IV node addresses to the X.25 DTE addresses. Multicast destinations (routing packets) are simulated by sending the packet to each X.25 DTE address in turn, following the order of the address translation list for that network. Use the X.25 **add address** command to configure the addresses.

To minimize multicast activity on an X.25 network increase the CIRCUIT HELLO TIMER parameter of NCP.

Routing

DNA IV handles both forwarding of DNA IV data packets and automatic routing with other DNA IV nodes. The router performs the following DNA IV functions:

- Announces its presence by sending hello messages on each network that has DNA IV enabled.

- Maintains a list of adjacent DNA IV nodes from the hello packets it receives from other DNA IV nodes.

- Exchanges routing information with other routers.

- Forwards packets between nodes.

All end and routing nodes periodically broadcast hello messages to the all-routers multicast address. This allows each router to know where every node in its area is, and how to get to a router in each area.

On each broadcast network (for example, Ethernet), one router declares itself the designated router for that wire. The designated router broadcasts its presence so that the endnodes know to use it as their default gateway. Any endnode sending a packet to a node not on that wire automatically sends it to the designated router for forwarding.

In a multi-area LAN, assign priorities to routers in such a way that the designated router is a level 2 router, or is likely to be the best next hop to commonly-used destinations. This reduces the possibility of traffic from endnodes having to take an extra hop to reach other areas.

Routing decisions are based on a least-cost algorithm. Each link (for example, point-to-point, broadcast network, hop) has a cost. Every router broadcasts (to other routers only) its cost and the number of hops to get to every node in its area. In this way, each router finds the cheapest cost path, subject to a maximum hop count.

Routing Tables

A router forwards any DNA IV data packet it receives to the proper node based on its routing table. To maintain its routing table, a router listens to and sends level 1 updates to every adjacent router in its area. If the router’s type is set to AREA, it also exchanges level 2 routing updates with adjacent level 2 routers.

Each router maintains a routing table with an entry for every node (up to the **maximum address**) and every possible next hop (all circuits and up to the **maximum broadcast routers**). Each entry in this table contains the cost and hop to reach a node through one circuit or next hop node.

Routing updates are sent out every **broadcast routing timer** seconds (10 by default). Updates are also sent when the router detects a topology change either in its own adjacency database or through an update from another router.

Area Routers

If the router is configured as an area router, it maintains a similar database for all of the areas up to **maximum area**, and can exchange area routing information with other area routers. Areas are handled almost exactly the same as nodes, except messages give costs to areas, but not nodes.

The areas concept results in two types of routing nodes:

- A level 1 router knows about only one area, so it keeps track of nodes in its area. Also, it cannot have adjacencies across areas (it ignores them).

- A level 2 router keeps an area routing database, and can have cross-area adjacencies. Level 2 routers advertise routes to all other areas, so level 1 routers send all foreign-area traffic to the level 2 routers.

Endnodes do no routing of their own, instead they pass packets to a router to be forwarded.

A level 2 router that can reach other areas advertises a route to node 0 within its area. When level 1 routers need to send a packet to another area, they route it toward the closest node 0 in their area. This is not necessarily the best route to that area, only to the closest level 2 router. From there, the level 2 routing algorithm sends the packet to its destination area.

Routing Parameters

In each system you can set the following routing parameters:

- Maximum number of nodes in the area.
- Maximum number of routers adjacent to this router.
- Maximum number of networks on any given node.
- Maximum number of endnodes one hop away from this endnode.
- Cost of a hop on each network to which this node is attached.
- Values of several timers involved in sending hello messages and expecting them from other nodes.

Access Control

Access control protects one group of nodes from other nodes on the network. Routers make all nodes on a network accessible to each other. Usually, the main forms of security are passwords and conservative use of DNA IV proxy access at the host level.

However, in some instances, the security level of machines may be so different that you might need to provide additional security by limiting access within the routers in the network. The DNA forwarder allows you to do this by using access controls.

Using Access Control

Generally, access controls are not recommended due to the following liabilities:

- Access controls affect performance of the router. Every packet is tested if you enable access controls.

- The more complicated the access control configuration, the greater the performance impact.

- Access controls are more difficult to configure, and errors in configuration can be difficult to diagnose.

- Access controls cannot hide a node from the routing protocols. It remains visible from all routers in its area, even if access controls prevent access.

Note: Access controls do not guarantee security; they only make intrusion more difficult. The DNA IV routing protocols used on Ethernet and other broadcast media have no security features in them, so finding ways to intrude is quite possible.

How Access Control Works

Access control prevents the forwarding of DNA IV (Long Format) data packets on the basis of source address, destination address, and interface. Access control does not affect routing packets, because they use a different packet format. This makes configuring access control safer, because you cannot break the routing protocol.

To implement access control, addresses are masked and compared. The address in question is masked with 1's in the bit positions to be tested, and 0's in the free area. The address is then compared to a fixed value. For example, if you use a mask of 63.1023 (all 1s) and compare it to a result of 6.23, it is true only for node 6.23. A mask of 63.0 and a result of 9.0, is true for any node in area 9.

These mask and compare values come in pairs for source and destination address. They are then formed into lists for an interface. Each interface can have one access control list, which is applied to packets received on that interface. This list may be inclusive or exclusive. An **inclusive** list is a set of address pairs that designates a corridor for traffic flow. An **exclusive** list is a set of address pairs that does not allow traffic flow.

In an inclusive list, the source and destination addresses are tested using the mask and compare values. If any entry’s source and destination matches, the packet is forwarded. In an exclusive list, the source and destination addresses are tested using the mask and compare values. If any entry’s source and destination matches, the packet is dropped. Make the choice between exclusive and inclusive on the basis of which list is shorter. However, exclusive access control is usually easier to configure.

When packets are dropped due to access controls, and if the Return to Sender Request (RQR) bit is set in the Long Format Data Packet header, the packet is returned to the sender. Then the connect request immediately fails, because NSP Connect Initiate packets are normally sent with the RQR bit set.

Configuring Access Control

Access control limits access to a particular host or group of hosts. You must assign access control to all routes to that host, not just the preferred route. Otherwise, access control functions as long as the primary route is up, but fails when the secondary route is in use.

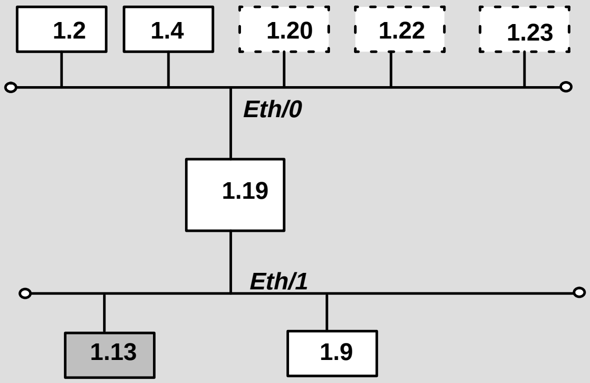
On your network map, draw a line to separate the secured region from the rest of the network. Ideally the line crosses the minimum possible set of adjacencies so that the least number of interfaces are running with access control. For broadcast networks (Ethernet), draw the line through the drop cable to the node, to identify the interface to filter. For each interface crossed by the access control line, define the same access control list using NCP.

Note: You do not need to define access controls in both directions. Because all DECnet applications use the NSP protocol, and the NSP protocol requires bi-directional connectivity, stopping packets in one direction is all that is required.

Inclusive Access Control

In the example in Figure 5–1, node 1.13 is the node that wants to communicate with nodes 1.2 and 1.4 only. Access control allows you to secure nodes from all nodes connected by routers. Therefore, in Figure 5–1 you can protect node 1.13 from all nodes except node 1.9 because these two nodes share the same physical network. To configure the desired access control for this example, build an inclusive filter on interface Eth /0 of router 1.19 as shown in the bottom of Figure 5–1.

Figure 5–1 Example of Inclusive Access Control



Inclusive filter information

Source result	Source mask	Destination result	Destination mask
1.2	63.1023	1.13	63.1023
1.4	63.1023	1.13	63.1023
0.0	0.0	1.9	63.1023

LKG–09896–95I

The first and second entries of the inclusive filter information shown in Figure 5–1 allow nodes 1.2 and 1.4 to send packets to node 1.13. The third entry allows any node to send to node 1.9 (you are not trying to secure node 1.9).

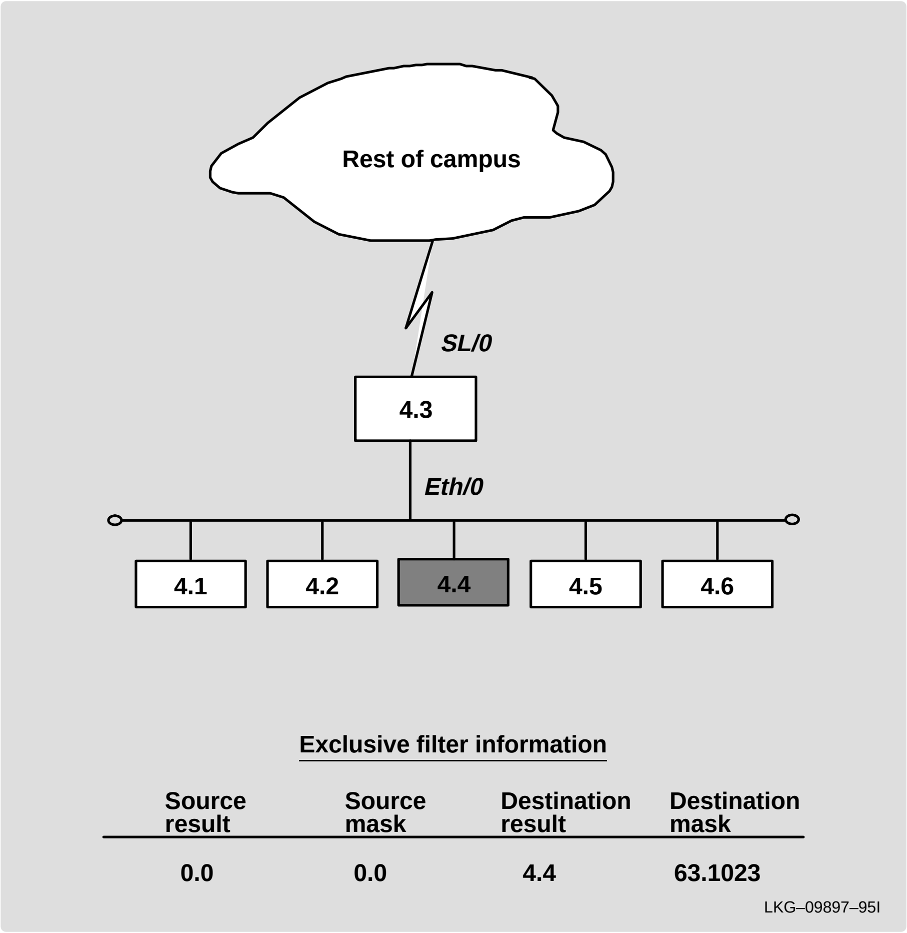
To configure the example given for router 1.19 in enter the following NCP commands and parameters:

```
NCP> def mod access-cont circ eth/0 type inclusive
NCP> def mod access-cont circ eth/0 filter 1.2 63.1023 1.13 63.1023
NCP> def mod access-cont circ eth/0 filter 1.4 63.1023 1.13 63.1023
NCP> def mod access-cont circ eth/0 filter 0.0 0.0 1.9 63.1023
NCP> def mod access-cont circ eth/0 state on
```

Exclusive Access Control

In the example in Figure 5–2, exclusive access control protects node 4.4 from the rest of the campus.

Figure 5–2 Example of Exclusive Access Control



Exclusive filter information

Source result	Source mask	Destination result	Destination mask
0.0	0.0	4.4	63.1023

Configure the desired access control for this example by building an exclusive filter on the SL /0 interface of router 4.3 as shown in Figure 5–2.

To configure the example given for router 4.3 in Figure 5–2, enter the following NCP commands and parameters:

```
NCP> def mod access-cont circ sl/0 type exclusive
NCP> def mod access-cont circ sl/0 filter 0.0 0.0 4.4 63.1023
NCP> def mod access-cont circ sl/0 state on
```

Area Routing Filters

Area routing filters allow special configurations of your DNA network. Because this is an advanced topic, very few DNA IV networks need routing filters. There are two primary applications for area filtering in DNA IV:

- Limiting access (security) to some group of areas from other areas.
- Allowing the blending of two DECnet address spaces.

Note: You must configure area routing filters carefully to avoid breaking your area routing. If you do not understand how DECnet routing works, especially at the area level, do not try to use routing filters. You can find more information about the DECnet routing protocol in *DECnet Digital Network Architecture Phase-IV Routing Layer Functional Description* (AA-X435A-TK).

How Area Routing Filters Work

Area routing filters allow you to configure a router to control the information about DECnet areas that are sent or accepted in level 2 routing messages. You may configure separate incoming and outgoing filters for each interface. Each filter specifies which area's routing information is passed to or accepted from.

When a network sends a level 2 routing update and there is a routing filter, the entry (RTGINFO) for any area not in the filter has the cost of 1023 and a hop count of 63. Any area in the filter has the correct cost and hops placed in the entry.

When the network receives a level 2 routing message and there is a routing filter, any entry for an area not in the filter is treated as if the cost is 1023 and the hop count is 63 (unreachable). Any routing entry from the packet that is in the filter is processed normally.

The routing filters affect the processing of level 2 routing messages only. There are no filters for level 1 routing messages. Routing filters have no effect on router hello processing, and do not prevent area routers from developing adjacencies with each other. They affect the area routing database only. If the filters prevent an area router from learning about any other area, they prevent the router from becoming attached. Then the router cannot advertise as an area router.

Security by Area Filtering

Like access controls, routing filters provide security. However, routing filters have some disadvantages compared to access controls:

- Area filtering is less flexible than access controls because it requires the assignment of areas to correspond to the desired security architecture.

- Area filtering is more difficult to understand and configure.

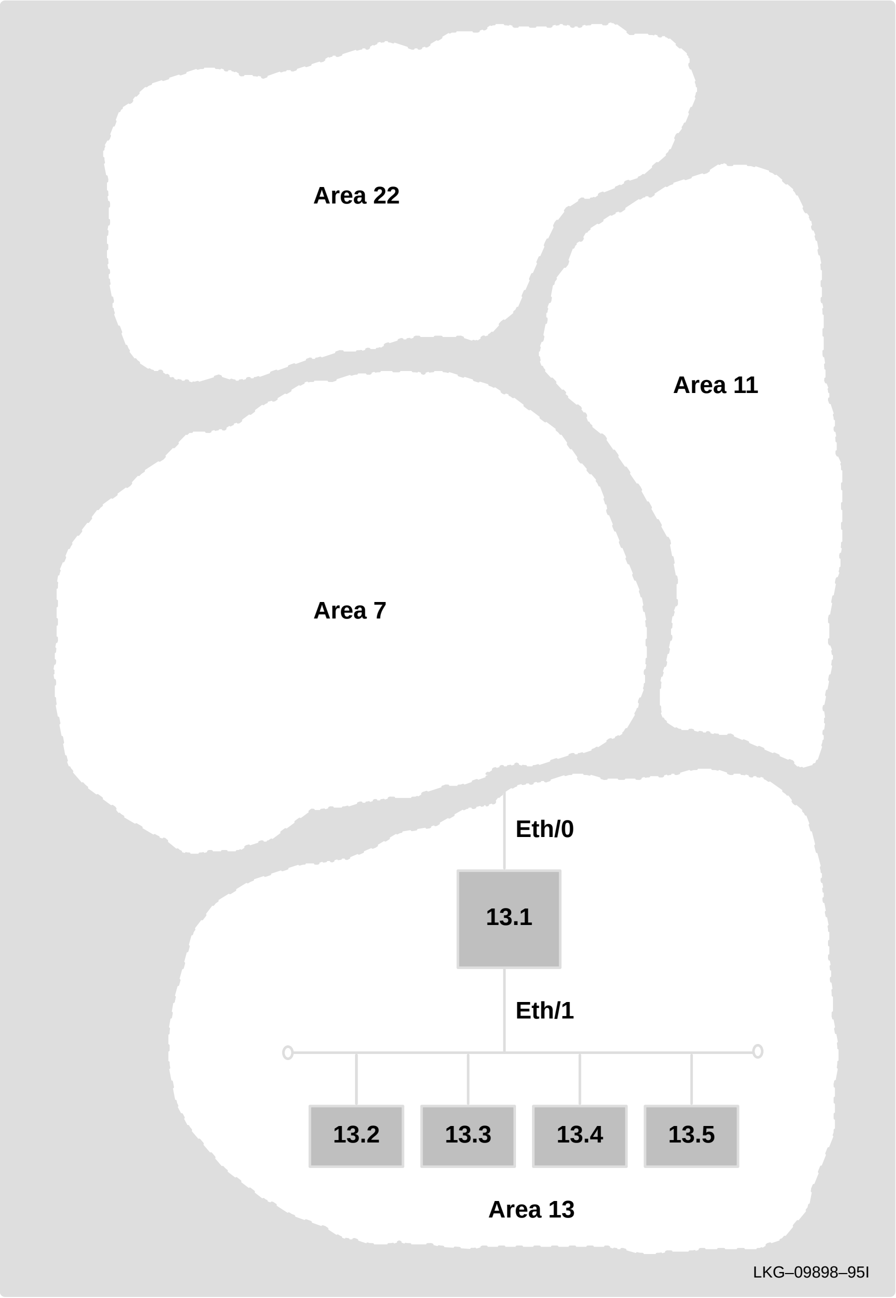
- The level of security is lower because a host that ignores the lack of routing information can send the packets to the correct router anyway.

However, area filtering is more efficient because there is no need to check every packet. In the following example area filtering occurs in an area that contains workstations that are part of a large network that contains machines with confidential information. There might be one machine outside their area that the confidential machines need to reach for information.

Figure 5–3 shows an example of using area routing filters to ensure security.

In Figure 5–3, area 13 contains workstations that need to be able to reach area 7. Node 13.1 is the router, and the other nodes are the workstations. Node 13.1 has a filter to accept only routes to area 7. Therefore, if node 13.1 receives a packet from any node in area 13 not destined for area 7, node 13.1 cannot forward the packet and sends the sending node an error message.

Figure 5–3 Example of Area Routing Filter for Security



To configure router 13.1 in Figure 5–3, enter the following NCP commands and parameters:

```
NCP> def mod routing-filter circ eth/0 incoming area 7
NCP> def mod routing-filter circ eth/0 incoming state on
```

Blending DECnet Domains

DECnet has a 16-bit node address space with a fixed hierarchy of 6 bits of area and 10 bits of node. By comparison, IP has a 32-bit node address space with a flexible multi-level hierarchy. Many established networks have now grown to the point where they use all 63 areas. The problem is that as different facilities connect to each other, they want to connect their DECnet networks but cannot due to area number conflicts.

Redesigning the DECnet architecture is the only solution of this problem. (This is addressed by DECnet Phase V.) However, by using area routing filters, it is possible to allow some overlap between two DECnet domains.

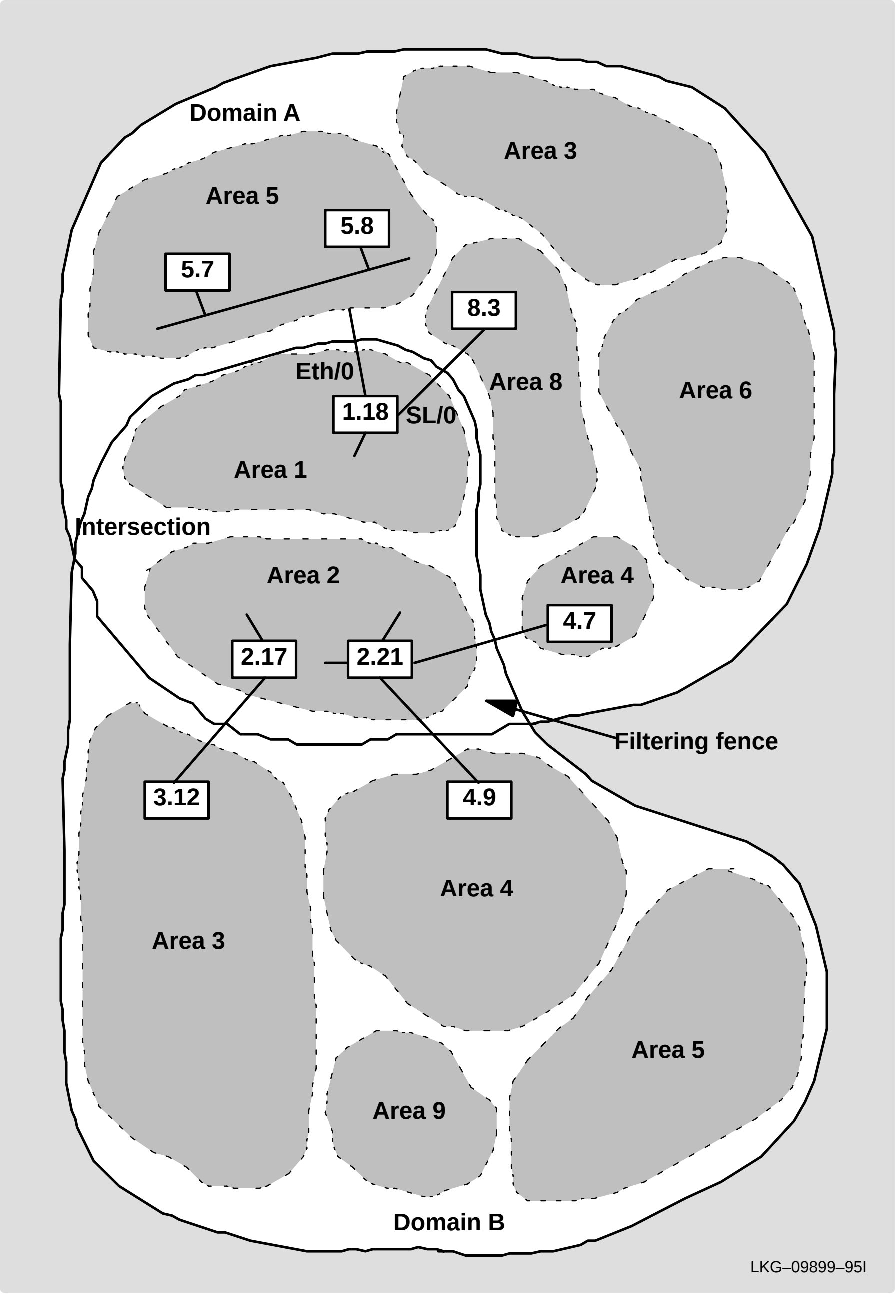
Domain is not a standard DECnet term; it is used here as a name for a DECnet wide-area network, presumably one with many areas. The goal is to blend two of these domains, so that there is a common area that can reach parts of both domains. However, there are more than 63 areas in the union of the two domains. Because area filtering is not simple to administer and is restrictive, do not consider using it if there are enough area numbers available for the union of the domains.

To configure the overlap of two domains, first you must decide which areas to intersect. These areas are the ones that are able to participate in both domains. These area numbers must not be used elsewhere in the two domains.

Figure 5–4 shows an example of blending DECnet domains.

In the example in Figure 5–4, the areas in the intersection are areas 1 and 2. The remainder of the areas can be duplicated between the two domains and are not in the intersection. In the example, there are two areas 3, 4, and 5, one in each domain. Note that it is never possible to allow direct connection between a node in area 3 in domain A and area 3 in domain B. The best that you can do is give the areas in the intersection the ability to talk to portions of each domain.

Figure 5–4 Example of Blending DECnet Domains



In designing the intersection, be careful that neither domain relies on routes through the intersection to maintain connectivity between areas that are not in the intersection. Since the routes in and out of the intersection are filtered, they probably do not offer normal reachability between all areas in the domain.

To decide how to configure the routing filters, draw a concise map of the configuration. On this map, locate all of the areas and outline the two domains. Then decide upon the filtering fence that you need to establish. Carefully go around the intersection of the two domains and locate all level 2 adjacencies that cross the filtering fence. These are one hop communications paths between level 2 routers that cross between areas.

In the example, there are six adjacencies that cross the fence, 1.18 to 5.7, 1.18 to 5.8, 1.18 to 8.3, 2.17 to 3.12, 2.21 to 4.7, and 2.21 to 4.9.

The first step in designing the area filters is to set up filters that keep the areas in one domain from being propagated into the other domain. The only area routes that leave the intersection are those for areas in the intersection. In the example, these are areas 1 and 2. Therefore, only routes for areas 1 and 2 are sent from nodes such as 2.17 and 3.12.

On point-to-point links such as 2.17 and 3.12, it does not matter which end filters, but it is probably safer to filter on the sending end. Therefore, there is a filter on the interface of 2.17 allowing forwarding only routes from areas 1 and 2. The same occurs on the two interfaces of 2.21 and the link from 1.18 and 8.3.

When the hop between two areas is an Ethernet or other broadcast media, such as 1.18 to 5.7 and 5.8, make the decision on another basis. Most Ethernets have most of the level 2 routing nodes in one area, and a few in the second area. Here, the filtering is on the few, rather than the many. In the example, node 1.18 is the interloper on the Ethernet in area 5, so it filters. Mode 1.18 sends routers only for areas 1 and 2 on the Ethernet.

You can filter on both ends of an adjacency. This adds an extra layer of security against accidental re-configuration. However, if both ends do not have filtering capabilities, then only one can filter.

Given these filters, the two domains cannot contaminate each other. However, for a node in the intersection, it is not clear which area 3 is reached when a connection is attempted to node 3.4. It depends on the current route and the circuit costs. Clearly, this is not ideal. It does not matter that there might only be a node 3.4 in domain A and not in domain B. Routing between areas is done solely on the basis of area; only the routers inside an area know the routes to nodes in that area.

You must establish a second set of filters to decide which instance of an area (domain A or B) is reachable from the intersection for each area not in the intersection. You may decide that nodes in the intersection could reach areas 3 and 4 in domain A and area 5 in domain B. In the example, this is done by configuring routers 1.18 and 2.21 to only accept routes to areas 3, 4, 6, and 8 from domain A. Routers 2.17 and 2.21 only accept routes for areas 5 and 9 from domain B.

Therefore, nodes in the intersection see a universe that contains areas 1 and 2 from the intersection, areas 3, 4, 6, and 8 from domain A, and areas 5 and 9 from domain B.

To configure router 1.18 in Figure 5–4, enter the following NCP commands and parameters:

```
NCP> def mod routing-filter circ eth/0 outgoing area 1-2
NCP> def mod routing-filter circ eth/0 outgoing state on
NCP> def mod routing-filter circ eth/0 incoming area 3-4,6,8
NCP> def mod routing-filter circ eth/0 incoming state on
NCP> def mod routing-filter circ sl/0 outgoing area 1-2
NCP> def mod routing-filter circ sl/0 outgoing state on
NCP> def mod routing-filter circ sl/0 incoming area 3-4,6,8
NCP> def mod routing-filter circ sl/0 incoming state on
```

There is still no way that a node in domain A area 5 can communicate directly to a node in domain B area 5. For nodes in these two areas to communicate, you must do a series of application level relays using the **set host** command. For example:

- Run the **set host** command to remotely login from a node in the domain A area 5 to a node in domain A area 8.

- Run the **set host** command to remotely login from a node in domain A area 8 to a node in area 1 or 2.

- Run the **set host** command to remotely login from a node in area 1 or 2 to a node in domain B area 5.

6

The OSI Protocol

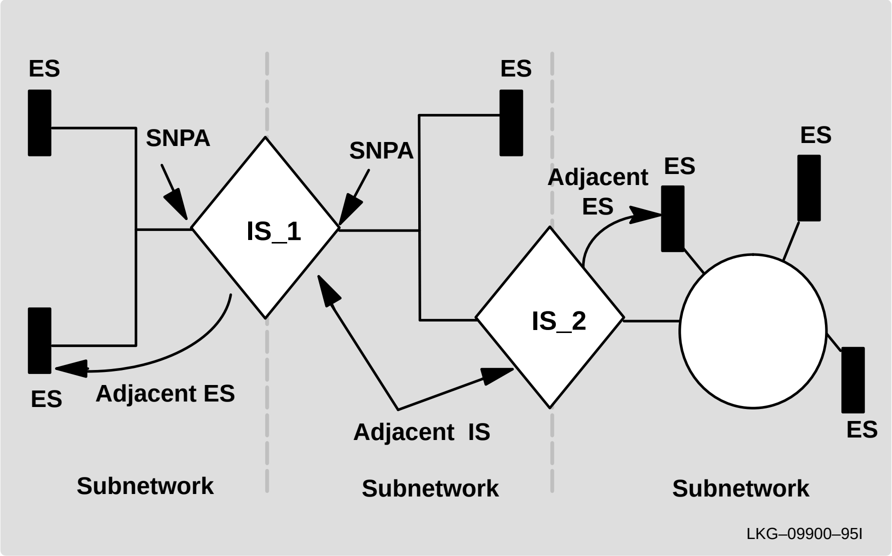
This chapter describes the router‘s implementation of the International Standards Organization’s (ISO) Open Systems Interconnection (OSI) Connectionless Network Layer. If you use DNA V networks, refer to this chapter for information about the use ISO protocols for DNA V.

OSI Network Overview

An OSI network consists of interconnected subnetworks. A subnetwork consists of connected hosts referred to as end systems (ESs) and routers referred to as *intermediate systems* (ISs).

Figure 6–1 illustrates an OSI network.

Figure 6–1 OSI Network



ESs contain all the layers of the OSI reference model and contain the host applications. ISs perform the functions of the lower three layers of the OSI reference model and handle the routing of the network protocol data units (NPDUs) between subnetworks. ISs logically attach to the subnetwork at the subnetwork point of attachment (SNPA). The SNPA is the access point into the data link layer.

Depending on the IS configuration, each IS can run three protocols: ES-IS, IS-IS (including Integrated IS-IS), and CLNP (Connectionless-Mode Network Protocol).

The ES-IS protocol allows the ESs and ISs attached to the same subnetwork to dynamically discover each other’s existence. An ES connected to the same subnetwork as an IS is *adjacent* to the IS.

The IS-IS routing protocol allows the ISs to do the following:

- Dynamically discover the existence and availability of adjacent ISs.
- Exchange routing information with other ISs.
- Use the exchanged routing information to calculate routes based on the shortest path.

The Integrated IS-IS extensions to the IS-IS routing protocol allow each IS to do all of the above and, in addition,

- Exchange IP route information with other IS’s and compute shortest paths for IP routes.

The CLNP protocol is a datagram protocol that transports packets between OSI systems.

NSAP Addressing

The NPDU contains OSI network addresses (also called NSAPs). The NSAP refers to a point at the network layer where the user accesses the network layer. NSAPs are unique points within a system that represent addressable endpoints of communication through the network layer. The number of NSAPs may vary from system to system.

An addressing authority, such as the United States government's National Institute of Standards and Technology (NIST), administers NSAP addresses and determines how the addresses are assigned and interpreted within their domain. If desirable, these authorities may further partition the domain into sub-domains and designate corresponding authorities to administer them.

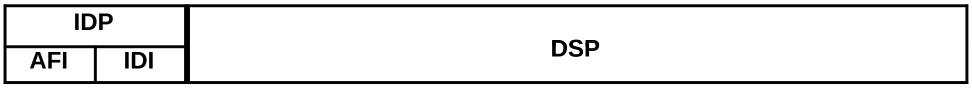
There are two NSAP addresses within the NPDU, a destination address and a source address. Each address can vary in length from 2 octets to 20 octets and is usually represented in hexadecimal notation. The following is an example of a NSAP that can be entered in the OSI configuration of the router.

490001AA000400010420

Because the address length is variable, portions of the PDU header called *Destination Address Length Indicator* and *Source Address Length Indicator* are used to indicate the length, in octets, of each address.

Figure 6–2 shows that an NSAP address consists of two parts: an Initial Domain Part (IDP) and a Domain Specific Part (DSP).

Figure 6–2 NSAP Address Structure



IDP

The IDP consists of two parts, the AFI (Authority and Format Identifier) and the IDI (Initial Domain Identifier).

The AFI specifies the type of IDI and the network addressing authority responsible for allocating the values of the IDI.

The IDI specifies both the network addressing domain from which the values of the DSP are allocated and the network addressing authority responsible for allocating values of the DSP from that domain.

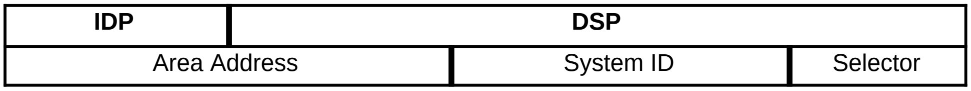
DSP

The network addressing authority identified by the IDI determines the format of the DSP. However, what is important is that the DSP includes addressing information specific to the domain.

IS-IS Addressing Format

Figure 6–3 shows how the IS-IS protocol divides the NSAP address into three portions: area address, system ID, and selector. The area address and system ID, together with a selector of 0, are referred to as a *Network Entity Title (NET*). An NET is the address of the network layer itself and is a value you configure into each IS in the OSI network.

Figure 6–3 IS-IS NSAP Addressing Interpretation



Area Address

In the IS-IS protocol, the area address includes all of the IDP and the portion of the DSP up to the system ID.

The area address is that portion of the NSAP that identifies a specific area within a domain. This address must be at least 1 octet long and all ESs and ISs in the same area must have the same area address.

If you are using DNA V, there are additional constraints that apply to the Area Address (see Chapter 7).

System ID

The system ID is that portion of the NSAP that identifies a specific system within an area. System IDs have the following attributes:

- Must be 6 octets long.

- Must be unique for each system throughout the area Usually the system ID is derived from a MAC address belonging to the system.

Selector

The selector is a 1 octet field that acts as a selector for the entity that is to receive the PDU, for example, the transport layer or the IS network layer itself.

GOSIP Version 2 NSAPs

Government Open Systems Interconnection Profile (GOSIP) Version 2 provides for government use the NSAP addressing format illustrated in Figure 6–4. The authorities responsible for the address have clearly defined the fields and specified the addressing format under the DSP set by the National Institute of Standards and Technology (NIST).

Figure 6–4 GOSIP Address Format

IDP		DSP						
AFI 47	IDI 0005	Ver 80	Auth.	Reserved	Domain (2)	Area (2)	Sys. ID (6)	Selector (1)

AFI This 1-octet field has a 47 (hexadecimal) designation. This value signifies that the address is based on the ICD format and that the DSP uses a binary syntax.

IDI This 2-octet field has a 0005 (hexadecimal) designation. This value is assigned to the U.S. government and the format is established by NIST.

VER This 1-octet field has designation of 80 (hexadecimal). This value identifies the DSP format.

Auth. (Authority) This 3-octet field identifies the authority that controls the distribution of the NSAP addresses.

Reserved This 2-octet field is provided to accommodate future growth.

Domain This 2-octet field contains the routing domain identifier.

Area This 2-octet field contains the area ID.

Sys. ID This 6-octet field identifies the system.

Selector This 1-octet field selects the entity to receive the NPDU.

Multicast Addresses

Datalink multicast addressing is the method that level 1 (L1) and level 2 (L2) ISs use to distribute link-state packets (LSPs), sequence number packets (SNPs), and hello messages to other systems on LANs. When an LSP, SNP, or hello message is multicast, a group of destination stations receives the packet. For example, an L1 LSP is multicast only to other L1 ISs.

You can configure multicast addresses for each subnet with the **set subnet** command.

Table 6–1 lists the multicast addresses for Ethernet and token ring LANs.

Table 6–1 IS-IS Multicast Addresses

Destination	Ethernet 802.3	Token ring 802.5	Address Description
All L1 ISs	0180C2000014	C00000008000	For all L1 intermediate systems on the subnetwork.
All L2 ISs	0180C2000015	C00000008000	For all L2 intermediate systems on the subnetwork.
All ISs	09002B000005	C00000008000	For all intermediate systems on the subnetwork.
All ESs	09002B000004	C00000004000	For all end systems on the subnetwork.

OSI Routing

OSI routes packets using the IS-IS protocol. Routing with the IS-IS protocol is based on one of the following:

- A system ID for routing within an area.
- An area address for routing within a domain.
- A reachable address prefix for routing outside the domain

The IS-IS protocol uses routing tables to forward packets to their correct destinations. The routing table entries are built from information in the link state database or from user-configured reachable addresses. The link state database is built from information received in the link state packets (LSPs). Refer to the section entitled “Link State Databases” later in this chapter.

IS-IS Protocol

The IS-IS protocol is a link state dynamic routing protocol that detects and learns the best routes to reachable destinations. IS-IS can quickly perceive changes in the topology of a domain, and after a short convergence period, calculate new routes. To accomplish this, the IS-IS uses the following packets:

- Link State Packets (LSP) that the IS uses to keep the link state database information current.

- Sequence Number Packets (SNP) to keep the database synchronized and to ensure that each adjacent IS knows which are the most recent Link State Packets (LSP) from each other router.

- Hello messages that ISs use to discover, initialize, and maintain adjacencies with neighboring ISs.

IS-IS Areas

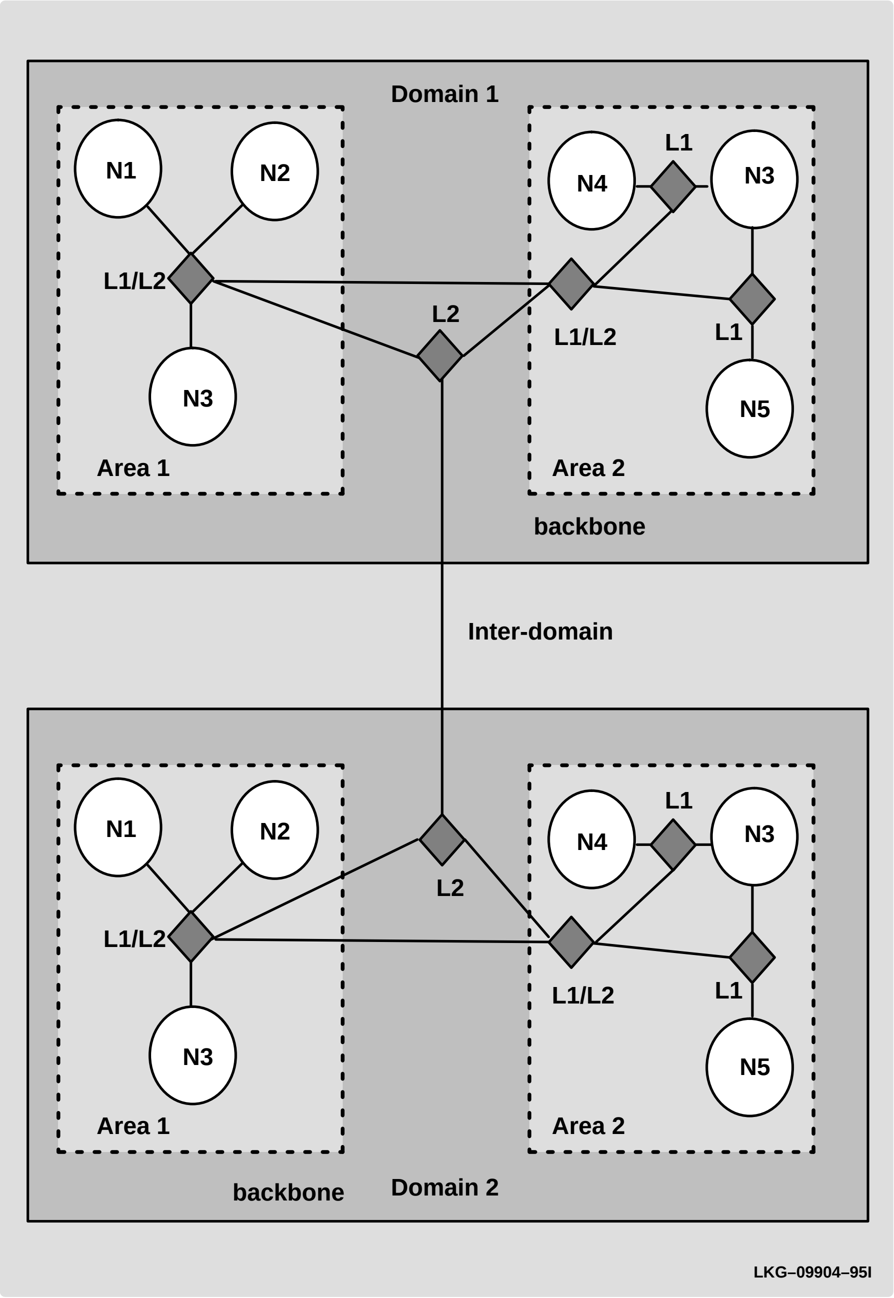
An IS-IS area is a collection of systems on contiguous subnetworks. Each area’s topology is hidden from that of the other areas to reduce routing traffic. A level 1 (L1) IS is used to route within an area. A level 2 (L2) IS is used to route between areas and as a level 1 IS within a single area.

Routing Domain

A routing domain is a single level 2 IS–IS backbone and all its attached level 1 Areas. Routing within the domain (“Intra-domain routing”) is done using the IS-IS protocol. Routing between separate routing domains (“Inter-domain routing”) is done using manually configured reachable address prefixes (“static routes’).

Figure 6–5 illustrates Intra- and Inter-domain routing.

Figure 6–5 OSI Domain



Synonymous Addresses

A single area can have up to 3 area addresses. These area addresses for the same area are called synonymous area addresses. There are two reasons why multiple area addresses are useful:

- To permit a router to have both an OSI area address and a DECnet Phase IV compatible area address.
- To aid migration from an old area address to a new area address. During migration the area can have both the new and the old area addresses.

IS to IS Hello (IIH) Message

The IIH message allows an IS to determine the existence of other ISs and to establish adjacencies. There are three types of IIH messages: L1, L2, and point-to-point.

The IS maintains a local hello timer for each interface and a holding timer for each adjacency. Each time the hello timer expires, an IIH is multicast over the IS's interface to any adjacent ISs. When the hello message is received, the recipient establishes or updates (refreshes) the adjacency information. This information remains current for amount of time (seconds) in the holding timer. If the holding timer expires, the adjacency is brought down.

The holding timer is derived from the hello timer received in the hello by multiplying it by a constant (the holding multiplier).

L1 IIH Message

The L1 IIH message is multicast over the interface when its local hello timer expires. The L1 IS places the following information in its IIH:

- Source ID.
- Any manual area addresses with which it is configured.
- IS type (L1 only, or L1/L2).

- Designated IS priority.
- LAN ID.
- If applicable, the system ID of the L1 designated IS (pseudonode).

Upon receiving this message, the adjacent L1 IS extracts the source ID of the sending IS. This IS then constructs its own IIH message and places its source ID into the source ID field. The sender's source ID is placed into the IS neighbors field. Returning the sender's ID verifies to the sender that the adjacent IS is aware that it exists.

When the first IS receives the IIH, it extracts the source ID and looks at the IS neighbor field. Upon discovering its own source ID in the IS neighbor field, this IS establishes an adjacency with the other IS. This ensures that ISs only form adjacencies with other ISs with which there is 2-way communication.

Note: Before the adjacent L1 IS can accept the packet, the packet must have a common area address with the adjacent IS.

L2 IIH Message

The L2 IIH is multicast over its interfaces for purpose of identifying itself to other L2 ISs. The L2 IS has the same functionality as an L1 IIH. The L2 IS places the following information in its IIH:

- Source ID
- Manual area addresses
- IS type (L2 only or L1/L2)
- Designated IS priority
- LAN ID
- If applicable, the system ID of the L2 designated IS

Point-to-Point IIH Message

A point-to-point IIH message is sent out over an IS's non-broadcast interface (Frame Relay or X.25) to identify itself to other ISs. A point-to-point IIH contains the following information:

- Source ID
- Manual area addresses
- IS type (L1 only, L2 only, or L1/L2)
- Local circuit ID

Designated IS

A designated IS is selected among all ISs connected to the same LAN to perform additional duties. In particular it generates link state updates on behalf of the LAN, treating the LAN as a *pseudonode*, a method of reducing the amount of information reported in link state packets.

When more than one IS exists on a LAN, each IS compares the following to determine which IS is the designated IS.

- All ISs compare their priorities. The IS with the highest priority becomes the designated IS.
- If the ISs have the same priority, they compare their source MAC addresses. The IS with the numerically highest MAC address becomes the designated IS for that LAN.

  The designated IS election occurs for each LAN interface. An IS with multiple LAN interfaces runs separate designated IS elections on each LAN and may be elected to be designated IS on multiples of its LAN interfaces.

Link State Databases

Each L1 and L2 IS contains a link state database.   The primary element of the database is the link state update (LSU). A link state update is the router’s internal representation of the set of LSPs received from another router. The router is responsible for building its own LSPs and processing other ISs’ LSPs to maintain the database. The L1 database contains information about system IDs in the IS’s area. The L2 database contains information about areas and reachable addresses. With information from the databases, the Dijkstra routing algorithm calculates the shortest paths to all destinations and builds the routing tables.

Link State Flooding

To ensure that each L1 and L2 IS contains the complete set of LSPs from all routers, LSPs are *flooded* throughout an area or a backbone. Flooding is a mechanism that an L1 or L2 IS uses to propagate an LSP to all L1 or L2 ISs. An L1 IS floods LSPs to L1 ISs only. An L2 IS floods LSPs to L2 ISs only. An L1/L2 IS floods both L1 and L2 LSPs.

L1 Link State Packet (non-pseudonode)

The L1 non-pseudonode LSP is flooded to all L1 ISs in the area. The L1 non-pseudonode LSP contains the following information:

- Source ID.
- Manual area addresses.
- IS type (L1).
- System IDs and costs of reaching IS adjacencies.
- If applicable, the system IDs of adjacent pseudonodes.
- System IDs for any manual ES adjacencies.

L1 Link State Packet (pseudonode)

The L1 pseudonode LSP is only generated by the L1 designated IS. It is flooded to all L1 ISs located in the area. Any L1 IS located on the same LAN that receives the LSP propagates the LSP to all L1 ISs adjacent on all of its other subnetworks. The L1 pseudonode LSP contains the following information:

- Source ID.
- IS type (L1).
- System IDs with zero cost for all ISs located on the LAN.
- System IDs and cost for any ES adjacencies learned through the ES-IS protocol on the LAN.

L2 Link State Packet (non-pseudonode)

The L2 non-pseudonode LSP is flooded to all L2 ISs. The L2 non-pseudonode LSP contains the following:

- Source ID.
- Set of area addresses for its Area (an L2 IS must be in exactly one area).
- IS type (L2).
- System IDs and the cost of reaching IS adjacencies.
- If applicable, the system ID of the pseudonode.
- Manually configured reachable address prefixes.

L2 Link State Update (pseudonode)

The L2 pseudonode LSP is only generated by the L2 designated IS. The L2 pseudonode LSP contains the following:

- Source ID.
- IS type (L2).
- System IDs with zero cost for ISs located on the LAN.

Attached and Unattached L2 IS

An *attached* L2 IS is a router that knows of other areas. An *unattached* L2 IS is a router that does not know of any areas other than its own.

When routing, an L1 IS routes packets destined outside its area to the closest attached L2 IS.

Routing Tables

An L1-only IS uses one routing table: the level 1 routing table. An L2-only IS contains three routing tables: an L2 area-address routing table, an L2 internal-metric reachable-address-prefix routing table, and an L2 external-metric reachable-address-prefix routing table. An L1/L2 IS contains the L1 routing table and all L2 routing tables. The routing table entries are built from information in the link state database.

L1 Routing

The following summarizes L1 routing:

1. An L1 IS receives a packet and compares the area address portion of the destination address in the header of the packet to the set of area addresses in the router.
2. If the packet is destined for the router’s area (that is, the destination address matches one of the router’s Area Addresses), the router extracts the system ID from the address. Searching for a match, the router compares the system ID to the system IDs in the L1 routing table.
3. If a match occurs, the IS routes the packet to the ES or the next hop L1 IS. If no match occurs, the packet is dropped.
4. If the packet is not destined for this area, the L1 forwards the packet to the nearest attached L2 IS or if this router is an L1/L2 IS, it checks its L2 routing tables as described in the next section. If the L1 cannot determine where to route the packet, the packet is dropped.

L2 Routing

The following summarizes L2 routing:

1. An L2 IS receives a packet and first tries to route it using L1 routing as described above. If the packed does not match any of the IS’s L1 Area Addresses, then the IS tries to route the packet using L2 routing.
2. The IS compares the destination address in the header of the packet to the set of area addresses (the set of Area Addresses of all Areas in the domain) in the area address routing table. If a match exists, the packet is forwarded to the next hop L2 router. If no match exists, the router checks the internal prefix routing table.
3. The internal prefix routing table contains entries of reachable address prefixes. If the internal routing table contains a match, the packet is forwarded to the next hop on the route to the prefix. If no match exists, the router checks the external prefix routing table.
4. The external routing table contains entries to reachable address prefixes whose metric type is external. If the external routing table contains a match, the packet is forwarded along the path to the prefix. If no match exists, the packet is dropped.

   Refer to the section “Internal and External Routing” for a detailed explanation of the internal and external routing tables.

Routing Metric

A routing metric is a value associated with the circuit to indicate the *cost* of routing over that circuit. For example, the routing metric based on the monetary expense of a circuit uses a low number to indicate a low monetary expense and high number to indicate a high monetary expense of routing a packet over that circuit.

The IS-IS routing protocol allows four routing metrics: default metric, delay metric, expense metric, and an error metric.

The current implementation of the OSI protocol uses the IS-IS default metric only. The default metric does not measure any defined property of the circuit. It is a dimensionless number that can be used by the network manager to affect the shortest path computation. All ISs in the routing domain must be capable of calculating routes based on the default metric. The other routing metrics are optional. Though they are not used by this implementation of the OSI protocol, they are described below for informational purposes only.

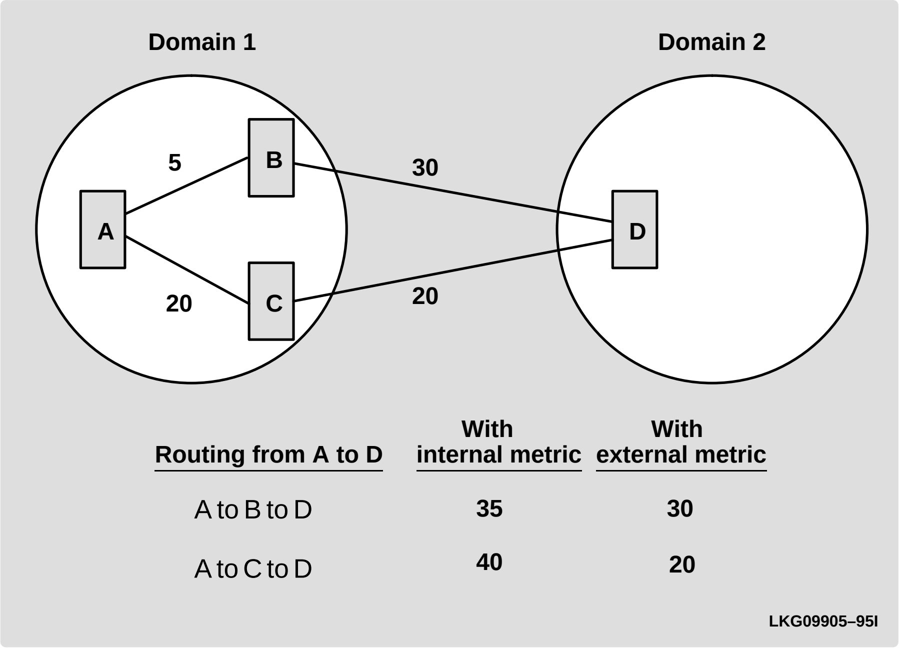
- The delay metric measures the transit delay of the associated circuit.
- The expense metric measures the monetary cost of utilizing the associated circuit.
- The error metric measures the residual error probability of the associated circuit.

Internal and External Routing

Internal or external routing involves an L2 IS routing a packet between two separate domains. When a packet needs to be routed to another domain, the L2 IS tries to match the address to a reachable address prefix in the internal or external routing table. Internal and external routes are based on the cost (routing metric) to the destination. An internal route's cost considers the cost of routing within the domain and the cost of routing to the destination. An external route's cost is based only on the cost of routing to the destination outside the routing domain. The IS chooses the path with the lowest cost.

Figure 6–6 illustrates L2 IS routing costs.

Figure 6–6 Internal and External Routing Metrics



Routing from A to D	With internal metric	With external metric
A to B to D	35	30
A to C to D	40	20

For example, in Figure 6–6 a packet is destined to go from node A in domain 1 to node D in domain 2. Node A can choose two paths to send the packet, to node B and then on to D or to node C and then on to D. How nodes B and C advertise the cost of their routes to D determines how node A decides to route the packet, internally or externally. There are four possible options:

- Nodes B and C advertise the cost of their routes to D as internal. The internal cost of the route A-B-D is 35, which is the cost of routing from A to B, plus the cost of routing from B to D. The internal cost of the route A-C-D is 40, which is the cost of routing from A to C, plus the cost of routing form C to D. Node A in this case chooses to route over the A-B-D path because the cost is lower.

- Nodes B and C advertise the cost of their routes as external. The external cost for A-B-D is 30, which is the cost of routing from B to D. The external cost for A-C-D is 20. Node A in this case chooses to route over the A-C-D path because the cost of this route is lower.

- Node B advertises the cost of its routes to D as external while node C advertises the cost of its routes to D as internal. Node A chooses to route over the A-C-D path because internal paths are always preferred over external paths regardless of metric.

- Node C advertises the cost of its routes to D as external while node B advertises the cost of its routes to D as internal. Node A chooses to route over the A-B-D path because internal paths are always preferred over external paths regardless of metric.

Note: Because there is no inter-domain routing protocol, all prefix routes between domains must be statically configured.

Address Prefixes

Routing at Level 2 is done by searching for the longest prefix route that matches the destination NSAP address in the packet to be forwarded. There are actually three different types of prefix route at Level 2:

- Area routes (prefix routes to other Areas within the ISIS routing domain)

- Internal Prefix routes (prefix routes that are to destinations outside the ISIS routing domain and whose metric is comparable to that used in the ISIS routing domain)

- External Prefix routes (prefix routes that are to destinations outside the ISIS routing domain and whose metric is not comparable to that used in the ISIS routing domain)

The Level 2 route search first looks for the longest matching Area route. If there is no matching Area route, the search looks for the longest matching Internal Prefix route. If there is no matching Internal Prefix route, the search looks for the longest matching External Prefix route. If there is still no match then the packet is dropped.

Representation of Prefix routes

It is important to understand how prefix routes are represented at the user interface and how they relate to the prefix routes that are transferred in the IS-IS protocol and used in the Level 2 routing tables.

It is also important to understand the encoding rules defined in ISO 8348 Addendum 2.

Table 6–2 Defined AFI Values and ISO 8348 Addendum 2 Rules

Table 6–2 shows the defined AFI values and the rules for their use.

Allocation Authority	AFI value	IDI value based on	Use AFI if IDI's leading digit is	Digits in IDI
Private	49	Not applicable	Not applicable	0 (none)
ISO DCC	39	Allocated for single-country organizations	Not applicable	3 (exact)
ISO 6523-ICD	47	Allocated for international organizations	Not applicable	4 (exact)
X.121	37 53	X.25 address	Non-zero Zero	14 (max)
F.69	41 55	Telex number	Non-zero Zero	8 (max)
E.163	43 57	Telephone number	Non-zero Zero	12 (max)
E.164	45 59	ISDN number	Non-zero Zero	15 (max)

The encoding rules for the preferred binary encoding defined in ISO 8473 Addendum 2 require that the IDI is always padded to its maximum length in an NSAP address. The digit value used to pad the IDI (if padding is necessary) is determined by the value of the most significant digit of the IDI. If that value is the digit 1, then the padding digits must be 0's. If the most significant digit of the IDI is 0, then the padding digits must be 1's. In addition, the AFI value must match the value of the padding digits as defined in table Table 6–2.

For example, suppose you have an E.163 based OSI addressing plan and you have one routing domain, A, all of whose addresses have the common E.163 number "123456789." Further suppose you have another routing domain, B, all of whose addresses have the common E.163 number "0987654321."

Now suppose that you have a routing domain, C, in which you use the IS-IS protocol and to which routing domain A attaches through router 1 and routing domain B attaches through router 2.

On router 1, you can create a single static prefix route (using the OSI configuration command **add route**) that makes all destinations in routing domain A reachable within routing domain C. Similarly, you can create a single static prefix route on router 2 that makes all destinations in routing domain B reachable in routing domain C.

The static prefix route on router 1 would be:

```
43000123456789
```

and that on router 2 would be:

```
57110987654321
```

In both these cases, you represent the prefix, including the padding digits, because the prefix ends logically at the end of the IDI field (you also include the padding digits if the prefix ends after the end of the IDI field; that is, within the DSP). However, if the prefix ends within the IDI (before its end), then you must not include the padding digits.

For example, suppose you also had another routing domain, D, whose addresses were based on two E.163 addresses 55512345 and 5559876. Also, suppose that you had a router, 3, that attached to routing domains D and C. On router 3, you can create a single static prefix route that provides reachability to all systems in routing domain D. This route's prefix is:

```
43555
```

This prefix does not include any padding digits between the AFI and IDI digits because the prefix logically ends in the IDI.

In addition, it is common for a prefix to end in the DSP (that is, after the end of he IDI). For example, a large corporation uses a single AFI and IDI value to generate globally unique OSI NSAP addresses for all systems in its network. To get the benefit of hierarchical routing within the corporation, it would use the leading digits of the DSP as routing domain identifiers.

In another example, an international organization that was assigned an ISO 6523–ICD value of 1234, and that has two routing domains, might use the prefixes:

```
4712340102
4712340203
```

for each of the domains. The AFI and IDI field in each of these is the same value: 471234. In these cases, the IDI field must be padded to its maximum length (although in this example there is no padding of ISO 6523–ICD IDIs because they are always allocated as 4-digit numbers).

There is another complication to prefix representation. This is the case of the ISO DCC and E.164 allocations. As Table 6–2 shows, these have an odd number of IDI digits. When encoded, the IDI field must be made an even number of digits in length. This is done by appending a trailing pad digit of hexadecimal value “f” at the end of the IDI digits. In the case of the E.164 allocation, the IDI digits may also need to be padded to the 15 digits length with leading padding digits to which the trailing “f” digit pad is then added. This trailing “f” digit pad is required only when the prefix logically ends at the end of the IDI or after the end of the IDI (within the DSP).

These rules describe the prefix representations you use and see at the user interface. However, destination NSAP addresses in packets that must be routed always contain the IDI field padded to its maximum length with the appropriate leading padding digits (according to Table 6–2) and with the trailing pad “f” digit, if necessary. The padding digits are effectively removed before searching for routes in the Level 2 routing tables.

For example, a packet destined to a system in routing domain A might have the destination NSAP address:

```
430001234567890102aabbccddeeff20
```

where 43 is the AFI specifying an E.163 address with zero digit padding, 000123456789 is the 12 digit padded E.163 address, 0102 is the first part of the DSP, aabbccddeeff is the 6 octet system ID, and 20 is the 1 octet selector.

A packet destined to a system in routing domain B might have the destination NSAP address:

```
57110987654321000456abcdefabcdef20
```

where 57 is the AFI specifying an E.163 address with “1” digit padding, 110987654321 is the 12 digit padded E.163 address, 000456 is the first part of the DSP, abcdefabcdef is the 6 octet system ID, and 20 is the 1 octet selector.

Default Address Prefixes

A default address prefix is used when you want to originate a default route to all addresses outside your domain. Default address prefixes are of zero length, so there is nothing to encode.

Authentication Passwords

To provide a minimum layer of security to the network, IS-IS provides the option of authentication passwords. When authentication is enabled, any IS-IS packet that does not contain the proper password is not accepted by the IS. The *authentication* field of the IS-IS PDU contains the authentication passwords.

The transmit password is added to IS-IS packets transmitted by the IS. The receive passwords are the set of passwords that the IS accepts. For example, with authentication enabled, if a transmit password is not added to the packet, or a listing of the transmit password is not in the receive password database, the packet is dropped. There are three distinct sets of transmit and receive passwords: *domain*, *area*, and *circuit*.

A domain password provides security for L2 routing information. An area password provides security for L1 routing information. A circuit password provides security for IS-IS hello messages.

ES-IS Protocol

The ES-IS protocol allows ESs and ISs attached to the same subnetwork to dynamically discover each other’s existence and availability. This information also permits ESs to obtain information about each other without an available IS.

Route redirection information allows an IS to inform an ES of a better route when forwarding NPDUs to a particular destination. For example, a better route may be through another IS on the same subnetwork as the ES, or the destination ES may be located on the same subnetwork as the source ES.

Hello Message

Addressing information is passed on to ESs and ISs through hello messages.

A local hello timer and a holding timer are present on each ES and IS. Each time the hello timer expires, a hello message is multicast on the LAN. When the hello message is received, the recipient sets its holding timer value by multiplying the value transmitted in the hello timer field of the message by its holding multiplier (usually 3). The recipient is expected to retain this information until the holding timer expires to ensure correct operation of the ES-IS protocol.

End System Hello (ESH) Message

The ESH message is multicast from the ES to all L1 ISs when its local hello timer expires. The ES constructs this message to inform an IS of any NSAPs that it serves. Upon receiving this message the IS extracts the system ID and SNPA information and stores the pair in its L1 routing table, replacing any other information currently stored there.

Intermediate System Hello (ISH) Messages

The ISH message is multicast to all adjacent ESs when its hello timer expires. The IS constructs this message to inform ESs of its NET. Upon receiving this message, the ES extracts the NET and SNPA information and stores the pair in one of its local routing tables, replacing any other information currently stored there.

Note: DNA V ESs are able to autoconfigure their NSAPs by extracting the Area Address field from the NET value of received ISH messages. The ES appends its own system ID (usually derived from one of its MAC addresses) to the Area Address together with one or more selector values to form a set of NSAPs for the ES. It then announces these in ESH messages it sends.

7

The DNA V Protocol

This chapter describes the recommendations for running the DNA V protocol and migrating from the DNA IV to the DNA V protocol. The DNA V protocol is based on the Open Systems Interconnection (OSI) protocols. All technical information concerning the OSI protocol is in Chapter 6, “The OSI Protocol.” All technical information concerning the DNA IV protocol is in Chapter 5, “The DNA IV Protocol.” This chapter assumes familiarity with DNA IV and OSI concepts and terminology.

DNA V Overview

The DNA V protocol adheres to the following International Standards organization (ISO) standards:

- **ISO 8473** – Connectionless mode Network Protocol (CLNP).
- **ISO 9542** – End System to Intermediate System routing exchange protocol (ES-IS).
- **ISO 10589** – Intermediate System to Intermediate System intra-domain routing information exchange protocol (IS-IS).
  - Including the Integrated IS-IS extensions defined in RFC 1095.

The DNA V protocol has the following features:

- **Backward compatibility** – The ability of DNA IV and V systems to exist within the same network and exchange data.

- **Selectable routing algorithm** – The ability to select the type of routing algorithm that allows the router to participate in both DNA IV and V areas.

DNA V Addressing

DNA V addressing complies with ISO 8348 Addendum 2. For more information about DNA V addressing, refer to Chapter 5.

DNA V Routing

Note: A DNA V IS can run either the IS-IS routing protocol link-state or phase IV routing protocol distance-vector at each of level 1 and level 2. Unless otherwise specified, throughout the rest of this chapter, assume that DNA V ISs are running the IS-IS routing protocol.

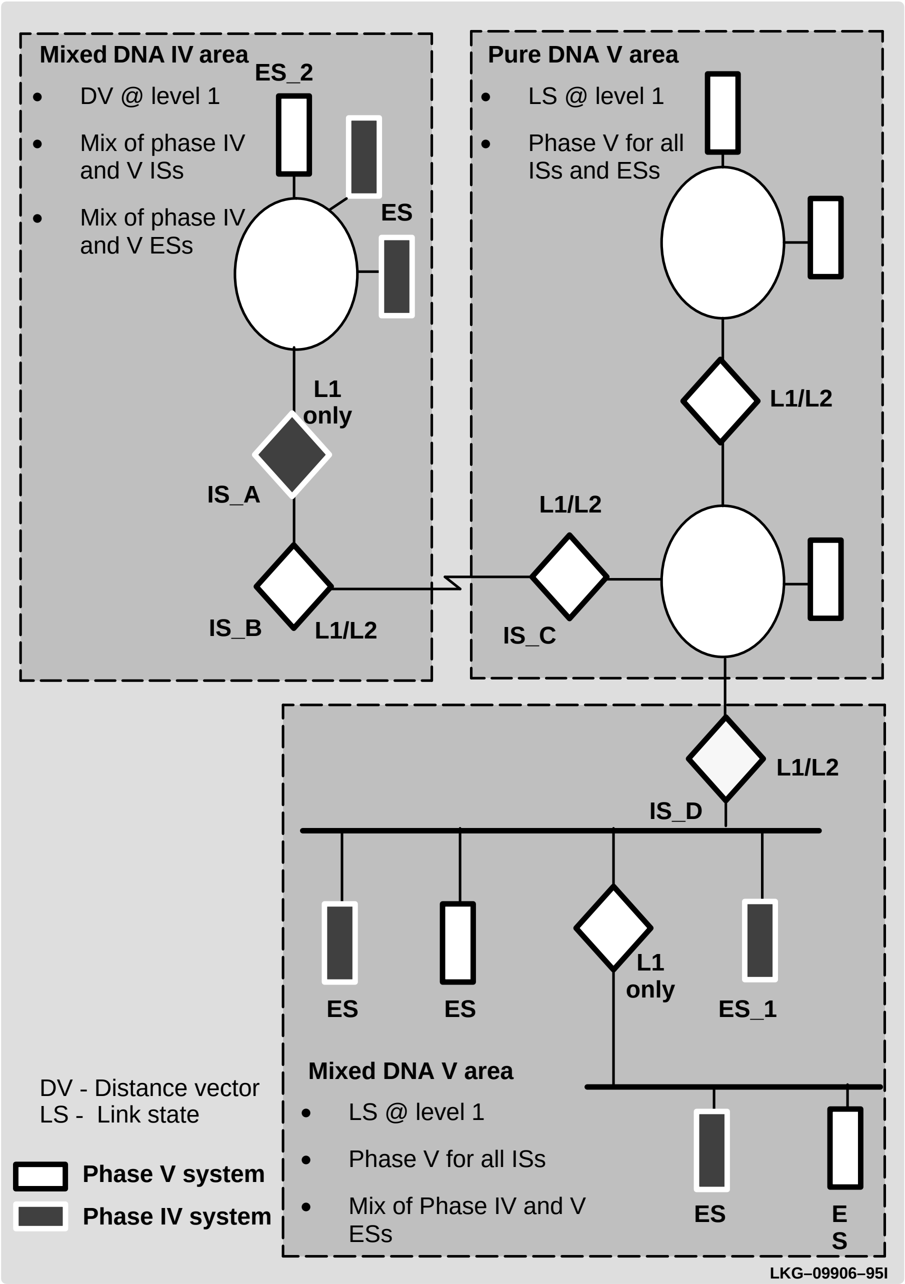
The IS-IS protocol is a *link-state* dynamic routing protocol that determines the shortest paths to reachable destinations. This information is then used to build the routing table entries. For further information about IS-IS routing, refer to Chapter 5.

DNA IV to DNA V Migration Strategy

This section lists the requirements for running a DNA V router in a DNA IV and V network environment. This section also discusses how to convert your DNA IV network to a DNA V network.

Figure 7–1 illustrates a DNA network running both DNA IV and V end systems (ES) and Intermediate systems (IS).

Figure 7–1 DNA IV and DNA V Network



Operating in DNA IV and DNA V Network Areas

You must consider certain requirements when operating DNA IV and DNA V systems (routers or ESs) in a mixed network. The following sections discuss these requirements.

Note: When configuring the DNA V protocol, you must use in the OSI configuration process. When configuring the DNA IV protocol, you must use the DNA configuration process.

IS Address Conversion and Network Layer Packet Translation

When routing between DNA IV and DNA V systems, the destination addresses contained in the network layer packets must be compatible with the routing protocol running on each system. A DNA V IS can translate a DNA IV packet containing a 16-bit address to a DNA V packet containing a Phase IV-compatible NSAP address and vice versa.

Phase IV Compatible NSAP Addresses

In a DNA V network, ISs can only translate DNA IV packets into DNA V (CLNP) packets and vice versa when the DNA V (OSI) address is phase IV compatible. A phase IV compatible address has the following format:

IDP	**DSP**			
Area Address	Loc-Area	System ID		Selector
		AA-00-04-00		
	2 octets	4 octets	2 octets	1 octet

The last 2 octets of the Area Address (which must be part of the DSP) have a value in the range from 1 to 63 (decimal) when interpreted as a 16-bit number. This number is interpreted as the phase IV Area Address. The first 4 octets of the system ID must be set to AA-00-04-00.

You must consider these rules when assigning the NET (NSAP) or Area Address to an IS. If you want the IS to be able to translate between phase IV and phase V formats, then the NET or one of the Area Addresses of the IS must be phase IV compatible and its Loc-Area value must match the phase IV area number to which it attaches.

Routing Algorithm

DNA V L1/L2 ISs run either the OSI link-state algorithm or the DNA IV distance-vector algorithm at each of level 1 and level 2. When configuring your router to run in a DNA network, use the **set algorithm** command to specify the type of routing algorithm that is running on the DNA interface.

Table 7–1 specifies the routing algorithm to use when operating in a specific DNA network type.

Table 7–1 DNA Network Routing Algorithm Selection

	Network Type		
Router Level	**DNA IV Only**	**DNA V Only**	**DNA IV/V Combination**
L1	Distance-vector	Link-state	Distance-vector or Link-state
L2	Distance-vector	Link-state	Distance-vector or Link-state

Routing Between Mixed Phase IV and V Areas

When routing within an area there are two restrictions:

- All ISs in the area must run the same routing protocol at level 1.

Note: The support for Phase IV cluster alias creates an exception to this restriction. Refer to “Phase IV Cluster Alias” for more information about this exception.

- All node addresses must be DNA IV translatable.

For example, in Figure 7–1, all the L1 ISs in the mixed DNA IV area are running the distance-vector routing protocol. All the DNA V systems in the DNA IV area have both a DNA IV 16-bit address and a DNA V NSAP address. To get a clearer understanding of how different routers operate with each other, follow a packet from ES_1 in the mixed DNA V area to ES_2 in the mixed DNA IV area as explained below.

1. A DNA IV packet is sent from ES_1 to IS_D.
2. IS_D reads the destination address, matches the address to an entry in the routing table, and translates the packet into a DNA V packet.
3. The DNA V packet is routed to the next hop, IS_C. IS_C reads the destination address, matches the address to an entry in the routing table, and sends the packet on to the next hop, IS_B.
4. IS_B reads the destination address, matches the address to an entry in the routing table, translates the packet into a DNA IV packet, and sends it on to the next hop, IS_A a phase IV router.
5. IS_A reads the destination the address, matches the address to an entry in the routing table, and forwards the packet to ES_2.

Phase IV Cluster Alias

A VAXcluster alias is an address that represents the VAXcluster. For a DNA Phase IV VAXcluster to have an alias, at least one of the member nodes must be a router. This router can only run the distance-vector routing protocol. If a VAXcluster has only one network interface, it can exist in an area in which the other routers are DNA Phase V and run the link-state protocol at level 1.

The link-state routers accommodate the VAXcluster router by:

- Detecting the alias in the distance-vector updates sent by the VAXcluster router.
- Creating a Phase IV ES adjacency to represent the alias.
- Advertising the alias in the L1 pseudonode LSP (of the L1 designated IS).
- Sending distance-vector updates to the VAXcluster router advertising the link-state router as a default route to all addresses in the area.

Migration Strategy

When migrating from DNA IV to DNA V, it is recommended that you do the following:

1. Switch the L2 ISs to run the same routing algorithm, either link-state or distance-vector L2.
2. Switch all ISs contained in a DNA V area to run the link-state routing algorithm at L1.
3. Switch all DNA V ISs in DNA IV areas to run distance-vector routing algorithm at L1.

8

The AppleTalk Protocol

This chapter describes the implementation of the AppleTalk protocol. The router uses separate protocols to support both the original AppleTalk (APL) and its recent enhancement AppleTalk Phase 2 (AP2).

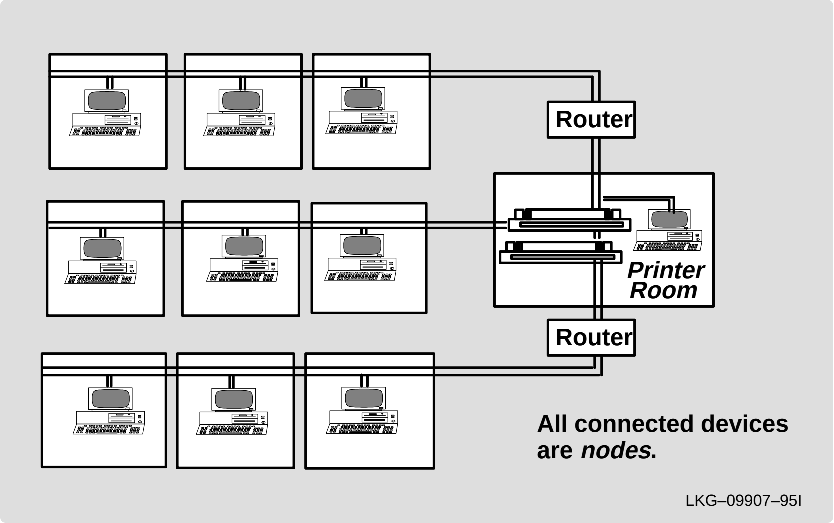
AppleTalk Overview

This implementation of the AppleTalk protocol is a collection of software components that transfers information between networks connected by physical media. AppleTalk operates by means of a distributed (client-server) network system. All users can communicate and share printers and files while interactions between users are transparent.

At the basic level of the typical AppleTalk network, connected devices are known as *nodes*. Most of the nodes are personal computers, but other nodes can be file servers, print servers, and routers.

Figure 8–1 illustrates the topology of a small portion of an internet (with connected nodes) that forwards packets through a router.

Figure 8–1 Typical AppleTalk Network Topology



At the next level are *networks*. A network is a group of nodes connected to a single logical cable. One or more networks then comprise groups known as *zones*. Zones are arbitrary (user-defined) subsets or conceptual groups of nodes within one or more networks. Groups of networks with defined zones connected by intelligent nodes (known as routers) comprise an *internet.*

AppleTalk's design allows you to include a variety of interface and cabling methods to build a network system. Routers interconnect these different interfaces to build even larger LANs or geographically dispersed internets. You can then choose different links for any portion of the AppleTalk internet according to the expected traffic, distance, and desired response characteristics in that portion of the internet.

Names

A name specifies the function of an entity in a network or an internet. Names are a universal set of machine identifiers represented in a meaningful, high-level form. Choose a name that is easily remembered by network and internet users. AppleTalk uses the Name Binding Protocol to map these human-readable names to machine-readable addresses. AppleTalk uses zone names and entity names.

- **Zone Name** – A name given by a network manager to an arbitrary subset of networks within an internet. This name is a string of not more than 32 characters. Any particular network belongs to only one zone. The networks in a particular zone do not need to be contiguous. The union of all zones is the internet. AppleTalk software stores the zone names in the Zone Information Table.

A network is not required to have a zone name. However, nodes and networks without zone names cannot advertise any services.

- **Entity Name** – In AppleTalk, an entity name is an 8-bit ASCII character string that has three fields: object, type, and zone. Each of these fields is a string not exceeding 32 characters in length. The object field refers to the type of service offered by the entity. The type field specifies the attributes of the entity. The zone name field identifies the location of the entity.

Node Number (node ID)

The node number is an 8-bit number that, when combined with the AppleTalk network number of a node, uniquely identifies each node on a network. Values 0, 254, and 255 are invalid.

AppleTalk Phase 1 and AppleTalk Phase 2

The original AppleTalk (Phase 1) was created for handling local device groups. Phase 1 handled a maximum of 254 concurrently active AppleTalk devices (nodes) on one network. Apple soon discovered that some large corporations were exceeding the limits of AppleTalk Phase 1, so they introduced AppleTalk Phase 2.

Phase 2 removes the original restriction of 254 devices on one network. You may now assign more than one network number to a single AppleTalk network by assigning a *network range*. This extended architecture theoretically increases the number of nodes per network to over 16 million, as well as providing an unlimited number of zones per cable.

Nonextended and Extended Networks

Networks running AppleTalk Phase 1 protocol (APL) are also described as *nonextended* networks. Networks running under AppleTalk Phase 2 protocol (AP2) are also described as *extended* networks. These networks differ in the number of devices supported and the network type supported.

Table 8–1 compares the differences between nonextended and extended networks in these areas.

Table 8–1 Number of Devices and Network Types Supported by Nonextended and Extended Networks

Feature	Nonextended Networks (AppleTalk Phase 1)	Extended Networks (AppleTalk Phase 2)
Number of Devices Supported	Maximum of 254 concurrently active AppleTalk devices (nodes).	254 or more concurrently active AppleTalk devices (nodes). Size of the range of network numbers determines the maximum number of concurrently active AppleTalk devices that can be supported on that network (253 devices per network number). The maximum number of concurrently active devices on an extended network equals the number of network numbers multiplied by the number of possible node IDs.
Network Types Supported	Nonextended AppleTalk Ethernet 1.0, serial line, or LocalTalk-based networks.	Typically EtherTalk 2.0, FDDI, and token ring-based networks that take advantage of the extended network configuration capabilities of Phase 2.

Nonextended and extended networks also differ in how each one handles AppleTalk node addresses assigned to its number of available devices (address resolution) and zones.

AppleTalk node addresses are 24 bits long. They are made up of two parts: a 16-bit network number and an 8-bit node number (node ID).   Zones are user-defined subsets or conceptual groups of nodes within one or more networks.

Table 8–2 compares how non-extended and extended networks resolve node addresses and handle zones.

Table 8–2 Address Resolution and Zoning in Nonextended and Extended Networks

Feature	Nonextended Networks (AppleTalk Phase 1)	Extended Networks (AppleTalk Phase 2)
Address Resolution	Nonextended network nodes (within a single cable) communicate using only their 8-bit node numbers. This is what gives the network its 254 device (node) limit.	Nodes in an extended (Phase 2) network communicate by unique network *and* node number pairs. Extended networks are also assigned a *range* of network numbers and all network numbers are chosen from within this range.
Zoning	Nonextended networks are assigned exactly *one* network number and *one* zone name.	Extended networks can have multiple zone names. Extended networks can be thought of as a group of nonextended networks, each residing on the same physical data link and capable of supporting up to 253 nodes (node ID $FE is reserved).

Note: When running a configuration that combines APL (Phase 1 nonextended) and AP2 (Phase 2 extended) networks, you may assign each network only one unique network number and one zone name.

Nonextended Networks and Address Resolution

Address resolution of node address on a nonextended network is an uncomplicated process since all nodes on the data link have one unique 8-bit node number (node ID). This means that the network only needs one network number to guarantee that all nodes on it have addresses that are unique in the Internet. The underlying data link protocol provides this unique node ID. You can then obtain the node’s network number is then obtained from a router using an Routing Table Maintenance Protocol Request packet. The following paragraphs explain this process in more detail.

AppleTalk implements dynamic address assignment. With this process, AppleTalk does not require that you specify all fields of an AppleTalk address when configuring of a router. If another preconfigured AppleTalk router appears on the network, it can be called on to supply the required network number for the new router. The preconfigured router, known as *seed router*, sends out the address information to all other routers on its connected network. The seed router is the one that comes up first and verifies the configuration of the other routers. If the configuration is valid, the other routers start functioning. The seed router comes up even if there are no other routers on the network. Routers that are not seed routers must first communicate with a seed router before they can function.

With dynamic addressing, a nonextended network device's node number is negotiated between AppleTalk hosts on the network (it may also be assigned by the network manager). AppleTalk automatically assigns node numbers or when a user-defined address is in use, it randomly selects an initial value.

The node first tries the node number that was its most recent address. If that value is not available, the node then searches for the next available address. If it reaches up to 254 without finding an available node number, it keeps returning to 1 until it finds a free address.

For non-seed routers, interfaces with enabled AppleTalk participate only in local routing until that interface's network number is determined. If zero was specified for a network number, that interface does not forward any packets until it receives a valid network number. Upon receiving a routing table update, the router is informed of the network number for the interface receiving the packet with the update. Every table update contains the network number of the network on which the packet was sent. Through this exchange, the router determines the network number of the receiving interface.

As long as one fully-configured seed router exists on the network, you do not need to configure the other interfaces and routers connected to that cable because they obtain their routing information from the seed. The seed router is configured with the network range and zone list while all other routers are given null values (for example, zero). Null values indicate that the router queries the network for values from the seed router. There are usually several seed routers on a network in case one of them fails. Also, a router can be a seed router for some or all of its network interfaces.

Extended Networks and Address Resolution

As mentioned, nodes in an extended AppleTalk network always communicate by network number and node number. When a router is not used, dynamic address resolution occurs by assigning a random network number within a network range as well as assigning a node number. Multiple zone names can be assigned to extended networks as well as network ranges. In this case, a node can access anything that is in any of the zones that are on the same cable as the node itself. The only provision is that a node can exist in only one zone and on only one network.

By adding a router to the network, a node starts up by using its newly obtained address for a short time. The node then requests a list of valid network numbers from the router(s). These numbers then select an unassigned address to obtain the actual AppleTalk address.

Configuration Considerations

The router provides separate protocols to support both AppleTalk Phase 1 (APL) and AppleTalk Phase 2 (AP2).   AP2 configurations for the network must have independent network numbers and zones names from the existing APL configurations.

To allow Phase 1 hosts to transparently communicate with Phase 2 hosts, you must enter the AppleTalk Phase 2 configuration process on the router running AP2 and enable the AppleTalk Phase 1/Phase 2 translation process through that router’s AP2 **enable translation** configuration command.

Besides providing the Phase 1/Phase 2 translation function, this router now acts as both a Phase 1 and a Phase 2 router on whatever interfaces these protocols are configured. The translation process passes routing information between Phase 1 and Phase 2 networks, resulting in a (logically) single internet.

The router supports AppleTalk networks that use 10 Mbit/sec. links, high-speed 80 Mbit links, and wide area links such as telephone lines that extend the geographical reach of the AppleTalk network. APL is currently routed on Ethernet, Serial Lines, and FDDI. AP2 is routed on all these media as well as on token ring 4/16.

Network Range (AppleTalk Phase 2 only)

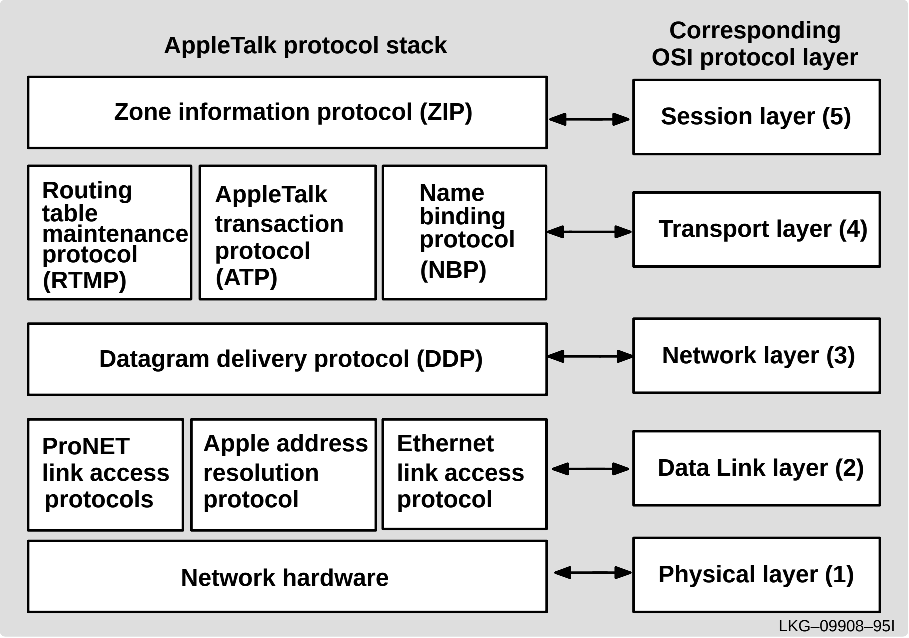
The network range is set of consecutive 16-bit network numbers. Each network within the AP2 internet is assigned a unique, non-overlapped network range. This is an extension of APL that uses a single 16-bit network number. Each user node on the AP2 internet has a unique address that is a combination of the network number and the node number. Each AP2 network can have a number of nodes that is equal to the product of net range and 253 (253 x *net range*), while an APL network is limited to 254 nodes.

The AppleTalk Protocol Stack

AppleTalk is a generic term that encompasses a large group of protocols. When grouped together, the AppleTalk protocols are referred to as a protocol stack. The AppleTalk protocol stack provides network access standards for layers one through five of the Open Systems Interconnection (OSI) reference model.

Figure 8–2 illustrates the AppleTalk protocols that correspond to particular OSI protocol layers.

Figure 8–2 OSI and the Corresponding AppleTalk Protocol Stack


AppleTalk protocol stack
Corresponding OSI protocol layer
Zone information protocol (ZIP)
Session layer (5)
Routing table maintenance protocol (RTMP)
AppleTalk transaction protocol (ATP)
Name binding protocol (NBP)
Transport layer (4)
Datagram delivery protocol (DDP)
Network layer (3)
ProNET link access protocols
Apple address resolution protocol
Ethernet link access protocol
Data Link layer (2)
Network hardware
Physical layer (1)
LKG–09908–95I

The specific subset of protocols necessary to forward AppleTalk packets include the following:

- Datagram Delivery Protocol (DDP), which is an AppleTalk protocol that resides in the OSI network layer.
- Routing Table Maintenance Protocol (RTMP), which is an AppleTalk transport layer protocol that maintains routing data.

AppleTalk also includes the following protocols:

- The Name Binding Protocol (NBP).
- Zone Information Protocol (ZIP).
- Echo Protocol (EP).
- A subset of the AppleTalk Transaction Protocol (ATP).

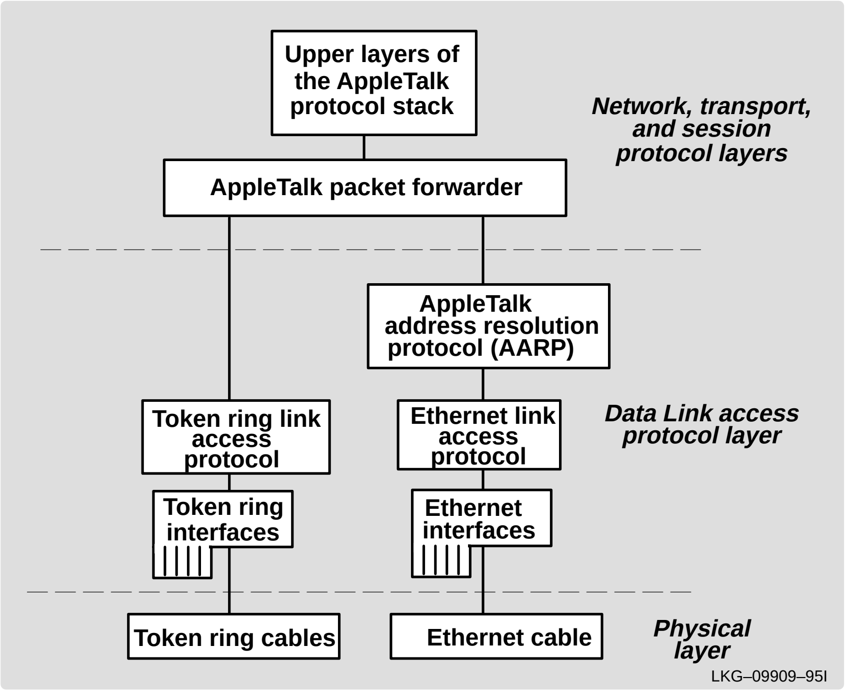
Data Link Layer Addressing

A data link layer protocol generally determines a hardware address. Each node on the network must have a unique hardware address, for example, a 48-bit Ethernet node address. Proper node addressing ensures that the network efficiently delivers and receives packets.

Using a router, you can transfer AppleTalk packets over a variety of networks including some developed networks and Ethernets.

Figure 8–3 illustrates where the data link protocols function in relation to the other layers of the AppleTalk protocol stack.

Figure 8–3 AppleTalk's Data Link Access Protocol Layer


Upper layers of the AppleTalk protocol stack
Network, transport, and session protocol layers
AppleTalk packet forwarder
AppleTalk address resolution protocol (AARP)
Token ring link access protocol
Ethernet link access protocol
Data Link access protocol layer
Token ring interfaces
Ethernet interfaces
Token ring cables
Ethernet cable
Physical layer
LKG–09909–95I

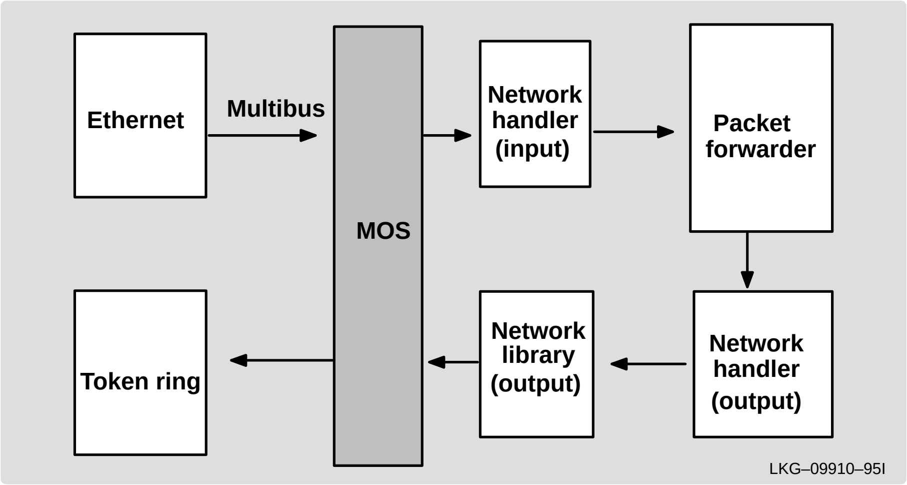
Ethernet node addressing is derived from the hardware address or from a user-defined initial address. When an Ethernet dynamically assigns a node address, the AppleTalk Address Resolution Protocol (AARP) confirms the uniqueness of that address or selects a new address. AARP maintains a set of protocol-to-hardware address mappings for each protocol stack that a node supports. The AARP stores and updates these address mappings in an address mapping table.

AppleTalk Packet Forwarding Concepts

AppleTalk packets on an internet are forwarded from network to network through a packet forwarder. The concept of packet forwarding is described in the following example.

Packet forwarding begins with the arrival of an AppleTalk packet on the Ethernet interface as shown in Figure 8–4. After the packet arrives on the interface, the device driver in the network handler receives the packet.

Figure 8–4 Path of AppleTalk Packet from Ethernet to Token Ring



Once the network handler receives the packet, the handler reviews the hardware header. Based on the value of the Ethernet type field, the handler passes the packet to the correct router software. Examples of router software include AppleTalk, IP, and DNA.

AppleTalk uses the destination information in the routing table to forward packets to the appropriate output network handler. The output network handler then passes the packet to the output network library that sends the packet out the associated device.

AppleTalk Packet Forwarding Protocols

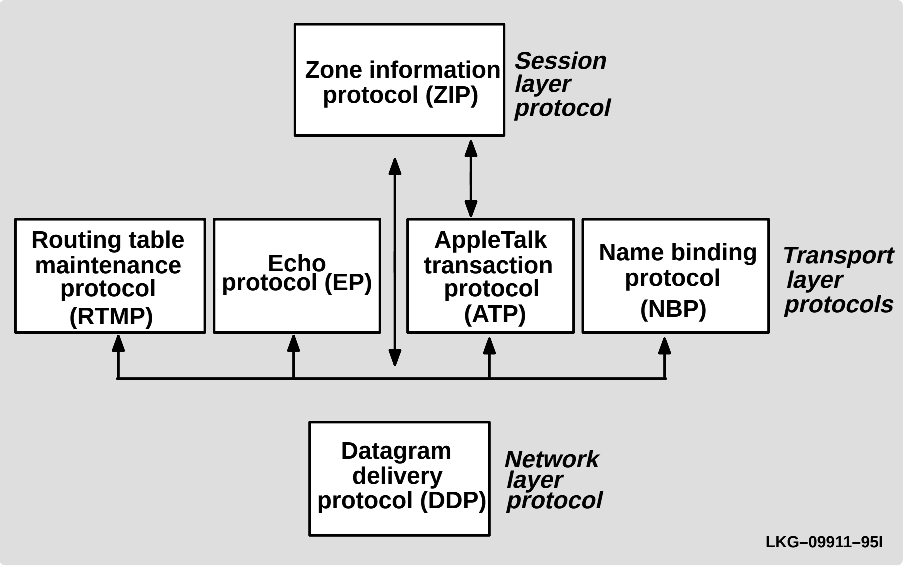
This implementation adheres to and supports the following AppleTalk protocols:

- Datagram Delivery Protocol.
- Routing Table Maintenance Protocol.

- Name Binding Protocol.
- Zone Information Protocol.
- AppleTalk Transaction Protocol.
- Echo Protocol.

Figure 8–5 illustrates the relationship between the AppleTalk protocols in the router’s AppleTalk protocol. Each of these protocols is described in Figure 8–5.

Figure 8–5 Implemented AppleTalk Packet Forwarding Protocols


Zone information protocol (ZIP)
Session layer protocol
Routing table maintenance protocol (RTMP)
Echo protocol (EP)
AppleTalk transaction protocol (ATP)
Name binding protocol (NBP)
Transport layer protocols
Datagram delivery protocol (DDP)
Network layer protocol
LKG–09911–95I

- **Datagram Delivery Protocol (DDP)** – The protocol follows this network layer protocol when combining a 16-bit network number, an 8-bit node address, and an 8-bit socket address to form a 32-bit internet address.

  You also have the option of specifying either short or long header DDP packets. If you configure your router to use short headers, packets generated by the router and destined for a directly-connected host use the short header DDP packets. Long DDP headers are the default value, as recommended by Apple.

- **Routing Table Maintenance Protocol (RTMP)** – The protocol follows this transport layer protocol when specifying how the AppleTalk protocol maintains the routing table for the entire internet.

  Every 10 seconds the protocol transmits packets with **good** and **suspect** entries to routing tables in connected routers. The routers receive these packets and update the corresponding entries to **good**. RTMP also specifies that every 20 seconds the protocol specifies the age of each entry in the routing table. After 20 seconds, entries with a state of **good** become **suspect** and entries with a state of **suspect** become **bad**. After an entry is **bad** for 20 seconds, the protocol deletes it.

  The protocol updates and maintains routing table hop counts based on the RTMP specifications. A hop count of 16 is considered infinite, and the protocol deletes all entries with this value.

  Hosts may send RTMP request packets to a router on an internet to learn the local network number. The router responds with an RTMP response packet.

- **Name Binding Protocol (NBP)** – The router follows this transport layer protocol when mapping entity names to protocol addresses for named entities on the internet. These network names are also associated with the type of service provided by the entity. Printing, file sharing, and mail service are examples of types of services. All NBP Broadcast Requests received by a router are transformed into one or more NBP lookup packets. The router forwards lookup packets as normal DDP data packets. The router does not have an entity name.

- **Echo Protocol (EP)** – Hosts follow this transport layer protocol when sending a request message to test whether a destination is reachable or not. The router responds to Echo Request packets with Echo Response packets.

- **Zone Information Protocol (ZIP)** – The router follows this session layer protocol when maintaining a Zone Information Table (ZIT). The ZIT lists the zone names and associated network numbers for connected AppleTalk networks. The network manager on any connected router may configure the zone name. Connected routers follow the ZIP specifications to learn zone names.

ZIP requires routers to transmit ZIP queries every ten seconds for each network in the routing table that does not have a zone name. If the zone name and network number mapping is known, the receiving router responds with a ZIP Reply packet. Using this query-reply method, eventually all the routers have identical information in their ZITs. Hosts obtain ZIT data by sending and receiving ATP packets to and from the router.

With AppleTalk Phase 1, you can also enable or disable ZIP takedown and bring up requests from hosts. These packets cause the router to remove or insert a routing table entry for a directly connected network. Removing or inserting an entry either brings up or takes down the network. This allows a host to change the zone name of a given network.

AppleTalk Tunneling

Digital’s implementation of the IP tunnel for AppleTalk Phase 2 is a single hop data link with no address or zone discovery functionality. No AARP is performed on the link because it is not numbered. Packets are encapsulated in a UDP (User Datagram Protocol) header (without a checksum) using port 748. The UDP contains the DDP packet only, without a special header. Normal RTMP’s are sent through the tunnel every 10 seconds.

UDP is a datagram-oriented protocol that transmits data packets for higher-layer protocols that do not require reliability. It does this without the overhead of TCP by limiting the services it provides. UDP does not perform error checking, does not acknowledge receipt of data, and does not sequence incoming messages. As a result, UDP messages may be lost, duplicated, or incorrectly ordered.

Higher-layer protocols pass data to UDP for delivery to same-layer processes. When UDP receives this data, it encapsulates it in segments with appropriate headers and passes the segments to IP. IP then encapsulates the UDP data in IP datagrams, determines the datagram’s destination path, and transmits the datagrams across the internet.

AppleTalk Data-Packet Format for IP Tunneling

An AppleTalk data packet that is forwarded across an IP tunnel by a router is preceded by the following protocol headers:

- A data-link header.
- An IP header.
- A UDP header.
- A domain header.

9

The Synchronous Data Link Control Relay

This chapter describes the implementation of the Synchronous Data Link Control (SDLC)/High-Level Data Link Control (HDLC) Relay.

SDLC Overview

SDLC Relay allows for the consolidation of SNA/SDLC traffic onto the corporate multiprotocol backbone. Compare Figure 9–1 to Figure 9–2.

Figure 9–1 shows a typical SNA installation using separate serial lines for SDLC and Source-Route Bridge Tunnel traffic.

Figure 9–1 Dedicated or Leased Line Configuration

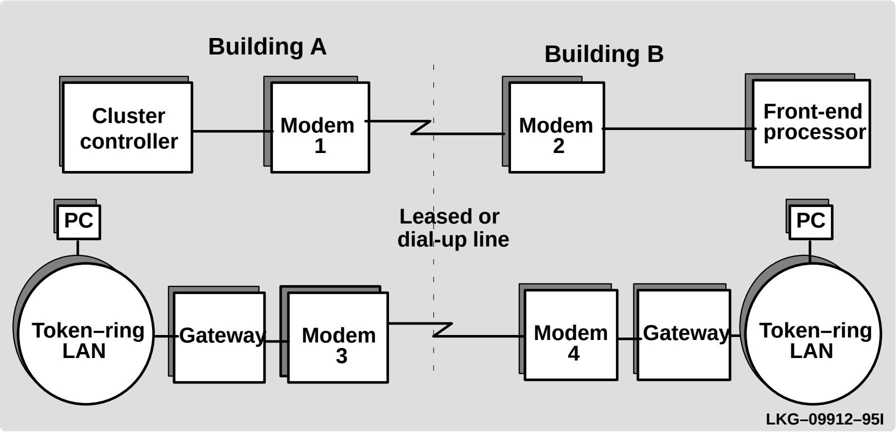
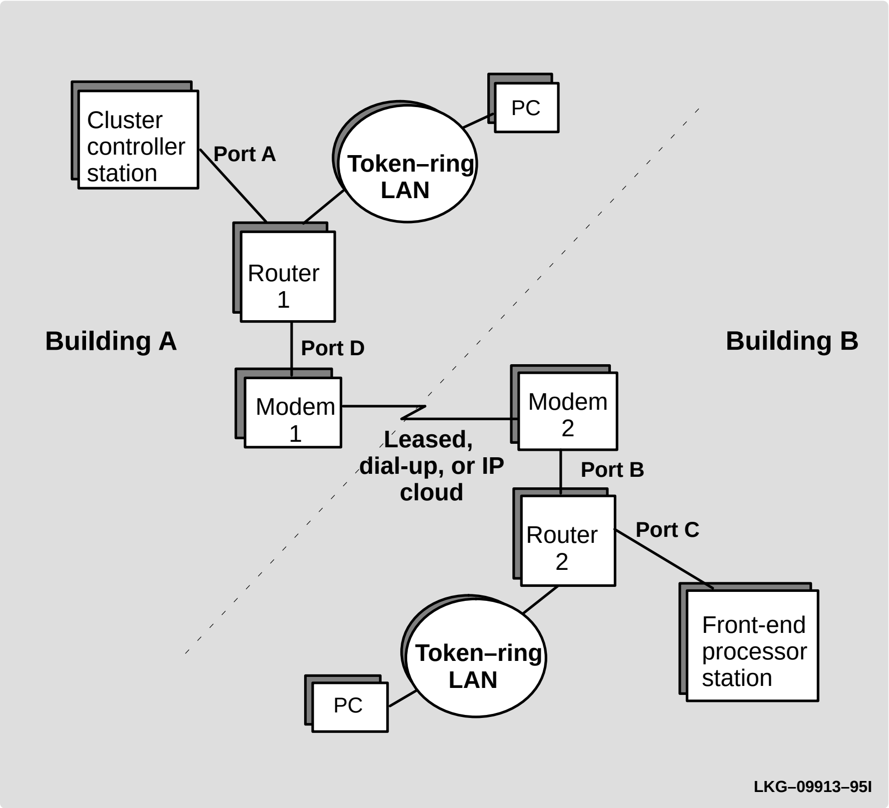


Figure 9–2 illustrates how the SDLC Relay reduces the WAN costs by consolidating the SDLC and LAN traffic across a common a serial link. In addition, the SDLC Relay allows the SDLC connections to take advantage of some of the benefits of a multiprotocol backbone, such as flexible configurations and dynamic routing using OSPF/MOSPF technology.

Figure 9–2 SDLC Configuration



Because SDLC traffic must now compete with non-SNA traffic for bandwidth on the backbone, terminal response times may vary when using dedicated serial links. To minimize these delays, you can utilize Bandwidth Reservation techniques. Refer to Chapter 13, for information about bandwidth reservation and priority queueing.

SDLC/HDLC Relay Device

The SDLC/HDLC relay device is physically the same as the serial line interfaces; however, when adding the device to the router configuration mode, you must add it as an SRLY (SDLC relay) device. Then the device can recognize that it is participating in SDLC/HDLC Relay.

The SDLC/HDLC relay device is a DTE (Data Terminal Equipment). This means that the SDLC/HDLC relay device does not supply a clock to the data line during transmit and must receive the clock from a modem eliminator (ME).

At other times, the SDLC/HDLC relay device is a DCE (Data Communications Equipment). This means that the SDLC/HDLC relay device does supply a clock to the data line during transmit.

Note: DO NOT configure any other local protocols on a previously configured SDLC device.

SDLC/HDLC Groups and Ports

For the SDLC/HDLC protocol to relay traffic, it must reference *ports* and *groups*. Ports refer to the devices added as SRLY devices. A group is made up of two ports.

SDLC/HDLC Forwarder

The SDLC/HDLC forwarder is responsible for receiving SDLC frames from the serial link, encapsulating them in UDP/IP packets, and transmitting them out the IP connection to the appropriate IP destination address. Conversely, IP packets received by the SDLC/HDLC protocol are first stripped of their UDP/IP header and are then transmitted out the SDLC link.

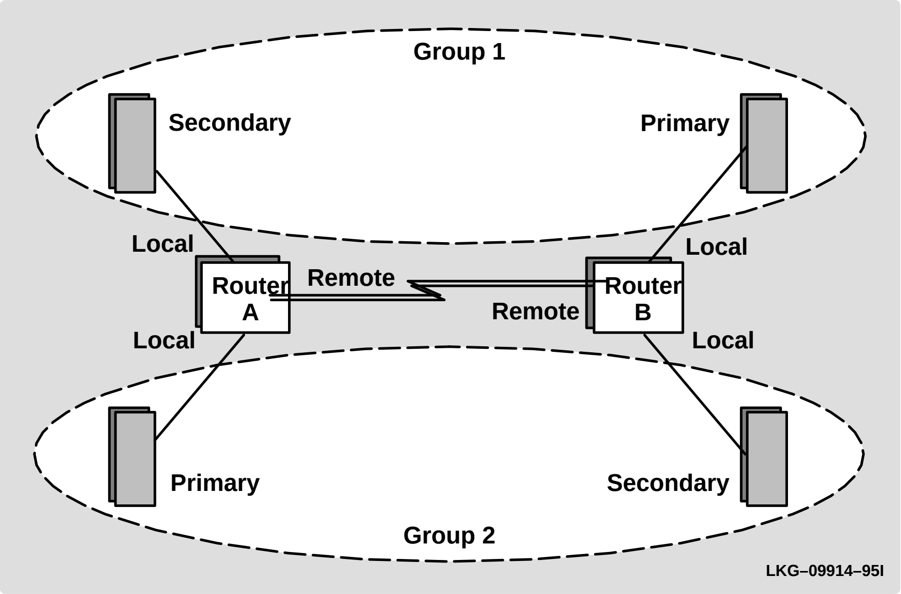
Since SDLC/HDLC Relay encapsulates SDLC frames within IP packets, IP routing can be used to route the SDLC traffic through any IP network. In addition, since every SDLC frame is encapsulated unchanged, this technique is completely transparent to the SDLC devices. Therefore, SDLC traffic generated by SNA PU types 5, 4, 2.0 and Node Type 2.1 are supported by SDLC Relay.

SDLC Primary and Secondary

The primary port on the router must be connected to the primary end station. The secondary port must be connected to the secondary end station.

The primary end station is responsible for management of the link, that is initiation, scheduling, and termination procedures. All transmissions are to or from the primary station. The secondary station must respond to the commands of the primary as shown in Figure 9–3. Use the **add local port** or **add remote port** command to configure a primary or secondary port.

Figure 9–3 SDLC Primary/Secondary and Local/Remote



SDLC Local and Remote

The SDLC local designation refers to the router interface that connects to the end station. A remote designation refers to the router interface that connects the IP internet link between two routers (Figure 9–3). When configuring a remote port, you must provide the IP address of the remote router.

SDLC Group

Every point-to-point relay SDLC/HDLC Relay session must have a unique SDLC group number associated with it. Because each point-to-point link can have only one primary and secondary end station, the group number ensures that the end stations communicate only with the end station it is intended.

System Network Architecture (SNA) Environment Information

To work in the SNA environment, you must be familiar with physical units (PUs) and logical units (LUs).

A PU represents an actual device connected to the SNA network, such as the cluster controller in Figure 9–2. A PU is responsible for the lower level communication between the device and the SNA network. To implement a PU, a combination of hardware and software is necessary, as well as unique identification numbers.

LUs represent a virtual logical path between users, providing a point of access through which they can interact with the SNA network. You can think of an LU as a socket that provides a point of access for each session associated with it. A device with one PU can have several LUs (depending on how many sessions are running on the PU).

PUs and LUs have time-out values and a number of retry values associated with them. After the modems establish a good connection, the secondary station receives polls from the primary station and responds to those polls. The number of retries is the number of times the primary continues to poll the secondary. When secondary fails to respond, the primary terminates the session and polling stops.

SDLC Relay does not implement poll *spoofing*. Spoofing filters polls are issued by the primary station. Not implementing spoofing allows all SDLC traffic to transfer transparently between the primary and secondary stations.

Line Topology and Discipline

This implementation of the SDLC Relay supports full-duplex transmission. You can use the SDLC/HDLC Relay in any point-to-point serial line configuration employing SDLC or HDLC at the data link layer. For example, you can use point-to-point links between a physical unit (PU) type 2 and a PU type 4 in an Systems Network Architecture (SNA) environment or between regular serial lines running SDLC/HDLC. An example of a PU type 2 is the cluster controller shown in Figure 9–2. An example of a PU type 4 is the front-end processor also shown in Figure 9–2. This topology supports dedicated SDLC Relay ports that allow no other functions to run simultaneously over the ports.

When configuring the SDLC Relay, you must inform each router of the presence of the other router. Both routers can have one local port and one remote port. The local port refers to the interface that is directly connected to the router through a modem eliminator (ME). In Figure 9–2, the local port for Router 1 is Port A, and the local port for Router 2 is Port C. The remote port refers to the virtual link that connects the far end station through the remote router. The SDLC Relay routes remote port traffic through the IP address of the remote router. In Figure 9–2, the remote port for Router 1 is Port B, and the remote port for Router 2 is Port D.

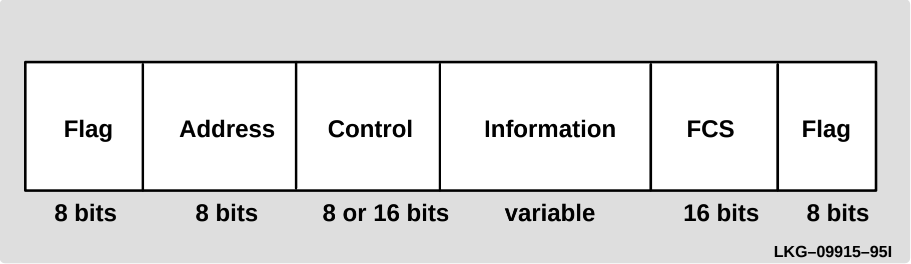
Note: The IP address of the remote port is entered when adding the remote port.

The SDLC Relay supports both schemes used to indicate that an SDLC link is idle. The two schemes are mark idle and flag idle. Mark idle refers to the lack of transitions on the data line during idle times. Flag idle refers to the presence of a flag pattern on the data lines during idle times.

Frame Structure

The SDLC Relay uses bit-oriented, synchronous transmission with a single-frame format for all data and control exchanges. Figure 9–4 shows the SDLC frame structure.

Figure 9–4 SDLC Frame Structure



Flag Fields

The field begins and ends each frame with a unique pattern of 01111110. Generally a single flag ends one frame and begins the next. All active stations attached to the link continuously search for the flag sequence to synchronize the start of the next frame. The sending and receiving stations use a process called *bit stuffing* to avoid the loss of synchronization due to the arbitrary appearance of the 01111110 bit pattern within the data stream.

With bit stuffing, the sending station adds a 0 after the fifth 1 in a non-flag bit stream. Then the receiving station monitors the bit stream. If the receiver detects a bit pattern of five zeros, it examines the sixth bit. If the sixth bit is a 0, the receiver deletes it. If the sixth bit is a 1 and the seventh bit is a 0, the receiver accepts the combination as the end of frame flag.

Address Field

The address field identifies the secondary station of the two member SDLC relay. This field is primarily needed in multipoint links, but is also used to preserve frame structure unity in point-to-point links.

Note: The SDLC/HDLC Relay ignores the address bits in the point-to-point configuration.

Control Field

The first one or two bits of the control field identify the SDLC frame type. SDLC Relay passes all types of frames. Frame types include the following:

- Information frames (I-frames) that carry data.
- Supervisory frames (S-frames) that carry error and flow control data.
- Unnumbered frames (U-frames) that provide supplemental link control.

Information Field

The information field contains the data that the frame transmits. This information field is present for all I-frames and only some U-frames.

Frame Check Sequence (FCS) Field

The frame check sequence field is a 16-bit cyclic redundancy check (CRC).

10

The X.25 Network Interface

The X.25 network interface connects a router to an X.25 packet-switched network.

X.25 Overview

The X.25 network interface software and hardware allow the router to communicate over a public X.25 network. The X.25 network interface complies with CCITT 1980 and 1984 specifications for X.25 interfaces by offering multiplexed channels and reliable end-to-end data transfer across a wide area network.

Refer to the *Network Interface Operations Guide* for information about the X.25 configuration and monitoring commands.

Protocol Handling

The X.25 interface provides both physical and logical access to remote X.25 hosts (routers). X.25 virtual circuits are either pre-established PVCs (Permanent Virtual Circuits) or dynamically set up SVCs (Switched Virtual Circuits) to remote destinations. The virtual circuit is a direct result of the router protocol packet forward request through the X.25 interface.

- **Permanent Virtual Circuits (PVCs)** – Permanent channels that remain connected after X.25 restarts. Because they are always present, these constant channels are similar to leased telephone lines. PVCs are suitable for high-volume data transfer and predictable protocol-specific traffic, such as routing updates. The X.25 protocol supports a maximum of 4 PVCs per port.

- **Switched Virtual Circuits (SVCs)** – Non-permanent channels that require Call Setup and Call Clearing. The temporary status of the SVC channel is similar to an ordinary telephone call: a connection is established, data is transferred, and the connection is terminated. SVCs are suitable for a low to medium volume of data transfer and bursts of protocol specific traffic like error correction.

If the destination protocol address maps onto a PVC or an existing SVC to the same X.25 destination for which the queue or window is not full, the packet is encapsulated into an X.25 data packet and forwarded to the X.25 network.

Failing to find an open virtual circuit and providing the router does not exceed the virtual circuit limits, results in an X.25 call to the matching X.121 network address. Subsequent protocol packets to the same destination are queued while waiting for completion of the X.25 call process. When the call is complete, all queued protocol packets are forwarded onto the network as X.25 data packets.

Configurable Parameters

The number of SVCs originated by the router is controlled by three factors:

- The maximum number of calls out. (See the **set calls-out** command for calls-out configuration information.)

- The default window size. (See the **add protocol** command for configuration information.)

The above limits prevent a flurry or burst of protocol traffic from consuming router or network resources.

As protocol traffic is queued at the network interface awaiting X.25 call completion, initially only one circuit is established to a destination. Once that circuit is established, additional circuits can be added to the same destination as needed. This is governed by both the configured window-size and calls-out parameters, and by the 4 or 10 circuit limit constraint given to a protocol destination. For example, if calls-out is set to 4 or greater and the protocol window-size is set to the default of 7, there is a potential of (window-size * 4-circuit limit) or 28 outbound protocol packets queued toward the destination network.

SVCs are terminated after a configurable period of idleness. The protocol idle-time period configuration overrides the global setting, thus allowing additional flexibility on a per protocol basis. Any additional circuits established during a burst of protocol traffic eventually clear, and protocol traffic eventually settles down and traverses on the earlier created circuits.

Applications lacking peer-to-peer keep alive mechanisms can be greatly affected by this idle-time period time-out feature. If protocol traffic is sparse, change the idle period to a reasonable time period to avoid unnecessary call clearing, but not so long as to hang unused circuits or to initiate a protocol time-out because of a lack of update messages. The default idle period is 30 seconds; a period of 90 seconds is a reasonable alternative.

Addressing

You must assign a unique X.121 network address to each X.25 network interface. Failure to set the network address prevents the X.25 interface from joining the attached network. When connecting to an X.25 switch this address must match the address of the switch configuration. When connecting to a public X.25 network, this X.121 address is assigned to your router by the owner of the X.25 network. The X.121 address is used by the remote DTE when establishing a call and by the router to identify itself when routing calls. The remote DTE maps the destination protocol addresses to the X.121 call addresses. The source address of one DTE is the destination for another, thus facilitating the piggy-backing of protocol return traffic on pre-established circuits. The mapping between the destination protocol address and the destination DTE address is configured using the X.25 **add address** configuration command. You may assign different protocol destination addresses to a single destination DTE address.

Mapping of the protocol to a X.121 call address is static (SRAM) and is configured on a per protocol and a per network interface basis. The exception is DDN addresses (IP HostTableFormat Addresses), which you can configure as static permanent entries or dynamically instantiated parallel to the IP protocol packet send sequence. Dynamic translations of IP HTF addresses to X.121 addresses are not saved over router restarts and are not displayed through the **list** option in the configuration command because they are not saved in SRAM.

For IPX and DNA IV, routing packets addressed to “all nodes,” X.25 sends the packet to each destination it knows about for that protocol (configured with the **add address** command), opening SVCs as necessary.

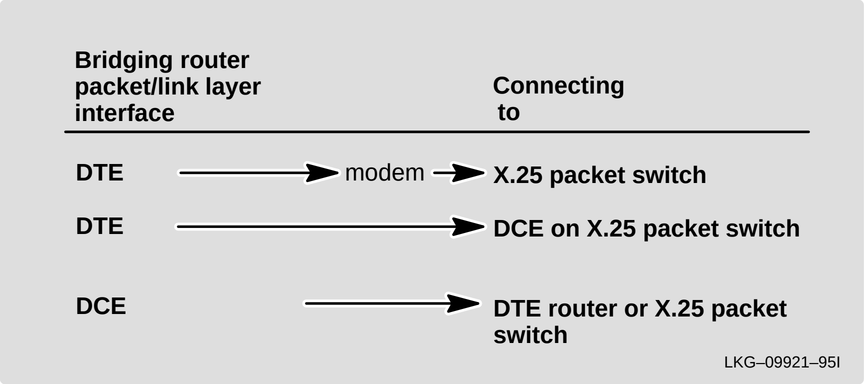
The X.25 Protocol Stack

The X.25 protocol stack provides network access standards for layers that are similar to the first three layers of the Open Systems Interconnection (OSI) reference model. The X.25 protocol stack consists of the physical, frame, and packet layers. These three layers closely resemble the physical, data link, and network layers of the OSI model.

The Physical Layer

The X.25 interface's physical layer specifies the hardware interface between the data terminal equipment (DTE) and the data circuit-terminating equipment (DCE). The X.25 physical layer specifies a cable interface that connects a DTE, such as the router to a public data network through a DCE device, such as a modem or DSU/CSU. Typical physical layer connections are described in Figure 10–1.

Figure 10–1 Typical X.25 Physical Layer Connections


Bridging router packet/link layer interface
Connecting to
DTE
modem
X.25 packet switch
DTE
DCE on X.25 packet switch
DCE
DTE router or X.25 packet switch
LKG–09921–95I

Though the router supports an X.25 DCE interface, DTE is the normal mode of operation.

The X.25 network interface software supports the following physical layer interfaces:

- RS-232-C at speeds up to 19.2-Kbps.
- RS-449 at speeds up to 64-Kbps.

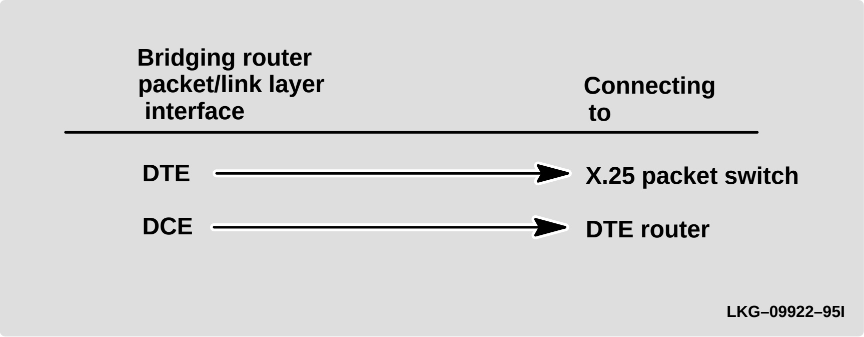
- V.35 at speeds up to 2-Mbps.
- X.21 at speeds up to 2-Mbps.

The Frame Layer

The X.25 interface's frame/packet layer, like the connection based data link layer in the OSI model, handles error control as the data travels on the interface between the router and the public data network. This layer, provides certain X.25 DTEs with an interface to the internet. When transmitting data from the router to the network, the frame layer supports the link access procedure-balanced (LAP-B) protocol.

Figure 10–2 shows the typical X.25 frame/packet layer connections.

Figure 10–2 Typical X.25 Frame/Packet Layer Connections



The Packet Layer

The X.25 interface's packet layer, like the network layer in the OSI model, establishes, manages and terminates end-to-end communications between local and remote hosts. The packet layer uses virtual circuit connections to establish communications between the router and a public data network with DCE capabilities. End-to-end communications issues include addressing, flow control, delivery confirmations, and interrupt signals.

X.25 software transfers packets using connection-oriented, or virtual circuit, packet switching. Virtual circuits transmit each packet sequentially, down a pre-established path. Virtual circuit connections are similar to placing a call over telephone facilities. Once the packet layer establishes a connection, the router can sequentially transmit data over the link.

Return traffic is usually directed on the same virtual circuit established from the source to the destination DTE, provided additional circuits are not open to the same destination.

11

The Frame Relay Network Interface

This chapter describes the Frame Relay network interface software.

Frame Relay Overview

The Frame Relay (FR) protocol is a method of transmitting internetworking packets by combining the packet switching and port sharing of X.25 with the high speed and low delay of time division multiplexing (TDM) circuit switching. FR allows you to connect multiple LANs to a single high-speed (1.54 Mbps) WAN link with multiple point-to-point permanent virtual circuits (PVCs). FR offers the following features:

- **High throughput and low delay** – Utilizing the core aspects (error detection, addressing, and synchronization) of LAPD datalink protocol, FR eliminates all network layer (layer 3) processing. Using only the core aspects the delay of processing each frame is lowered.

- **Congestion detection** – Upon receiving Backward Explicit Congestion Notification (BECN) the router initiates a controlled slow down of traffic to the CIR (Committed Information Rate), avoiding a complete FR network shutdown.

- **Circuit access and control** – As the router dynamically learns about the availability of non-configured circuits, you can control access to those new circuits.

- **Network management option** – As your network requires, the FR protocol can operate with or without a local network management interface.

- **Multiplexing protocols** – Using one PVC to pass the multiple protocols.

FR provides no error correction or retransmission functionality. To provide error free end-to-end transmission of data, FR relies on the intelligence of the host devices.

Frame Relay Network

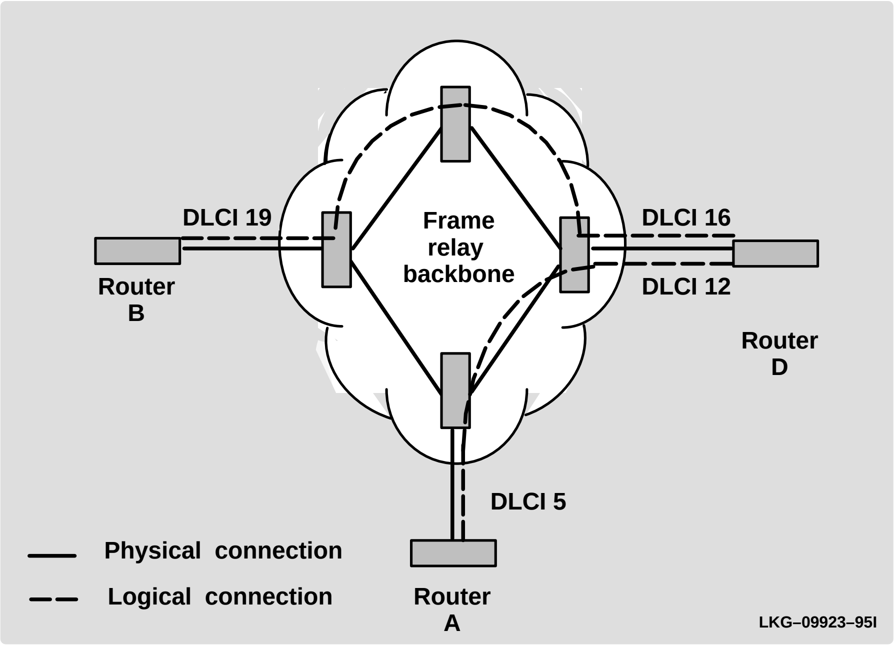
The Frame Relay Network comprises the backbone that provides the FR service. This network consists of FR switches that are provided by the FR service. The router functions as the FR connection device.

The router encapsulates FR frames and routes them through the network based on a Data Link Connection Identifier (DLCI). The DLCI is the MAC address that identifies the PVC between the router and the FR destination device.

For example, In Figure 11–1, a packet destined to go from router B to router D has a DLCI of 19 to reach router D. A packet destined to go from router D to router B has a DLCI of 16.

Figure 11–1 shows DLCIs in a frame relay network.

Figure 11–1 DLCIs in FR Network



A DLCI can have either local or global significance. Local DLCIs are significant at the point of entry to the network, but global DLCIs are significant throughout the network. To the user however, the DLCI that the router uses to route a packet is the DLCI that the user associates with the frame's global or local destination. DLCIs are configured through the FR configuration process or learned through FR management.

An FR network has the following characteristics:

- Transports frames transparently; the network can modify only the DLCI, congestion bits, and frame check sequence. HDLC (High-level Data Link Control) flags and zero bit insertion provide frame delimiting, alignment, and transparency.
- Detects transmission, format, and operational errors (frames with an unknown DLCI).
- Preserves the ordering of frame transfer on individual PVCs.

- Does not acknowledge or retransmit frames.

Frame Relay Interface Initialization

The FR interface is up when successful interaction with Local Management Interface (LMI) occurs; however, no data can be received or transmitted until an active PVC status appears through full status messages.

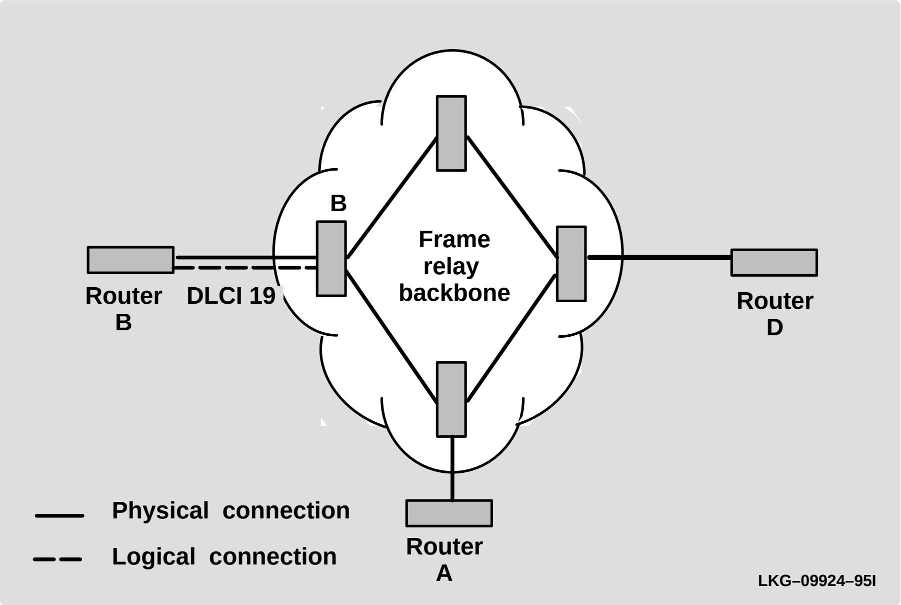
PVC status appears for all PVCs as either active or inactive. An active PVC has a completed connection to an end system.

An inactive PVC does not have a completed connection to an end system because either an end system or an FR switch is off-line.

For example, in Figure 11–2 router B has a configured PVC to router D. Router B is successfully interacting with FR management through FR switch B. Because either another FR switch is down or the end system is down, the end-to-end PVC connection is not established. Router B receives an inactive status for that PVC.

Figure 11–2 shows DLCIs in a frame relay network.

Figure 11–2 DLCIs in FR Network



Orphan Circuits

An orphan circuit is any PVC that is not configured for your router, but is learned indirectly through the actions of the network management entity.

Figure 11–3 assumes that router B has a configured PVC to router D, but none to router A. Router A configures a PVC to router B. Router B then learns about the PVC to router A and classify it as an orphan.

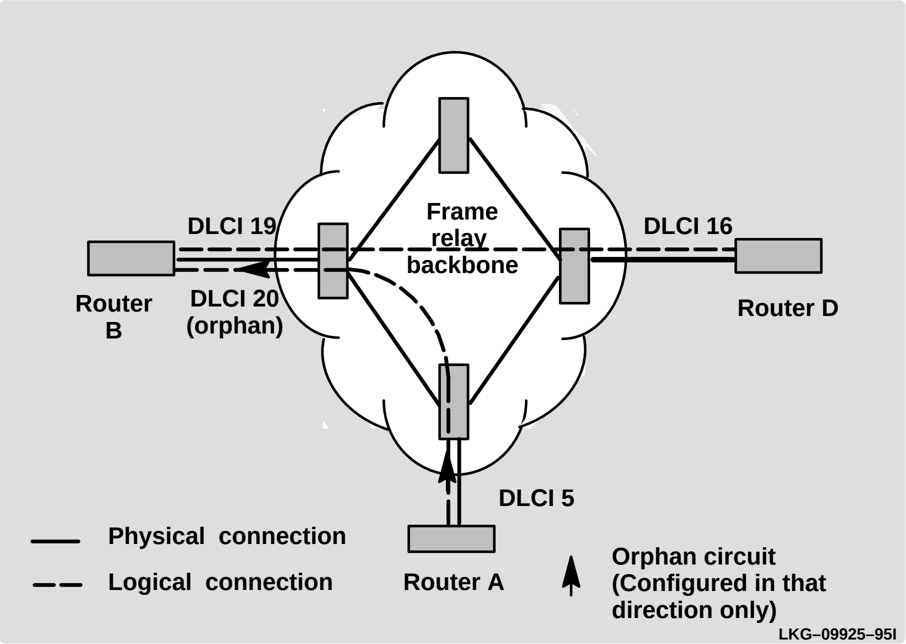
Orphan circuits are treated the same as configured circuits except that you may enable or disable their use with the **disable** and **enable** commands.

By disabling orphan circuits, you add a measure of security to your network by preventing any unauthorized entry into your network from a non-configured circuit.

By enabling orphans, you allow the router to forward packets over circuits you did not configure. Packets that are normally dropped are now forwarded.

Figure 11–3 show a sample orphan circuit.

Figure 11–3 Orphan Circuit

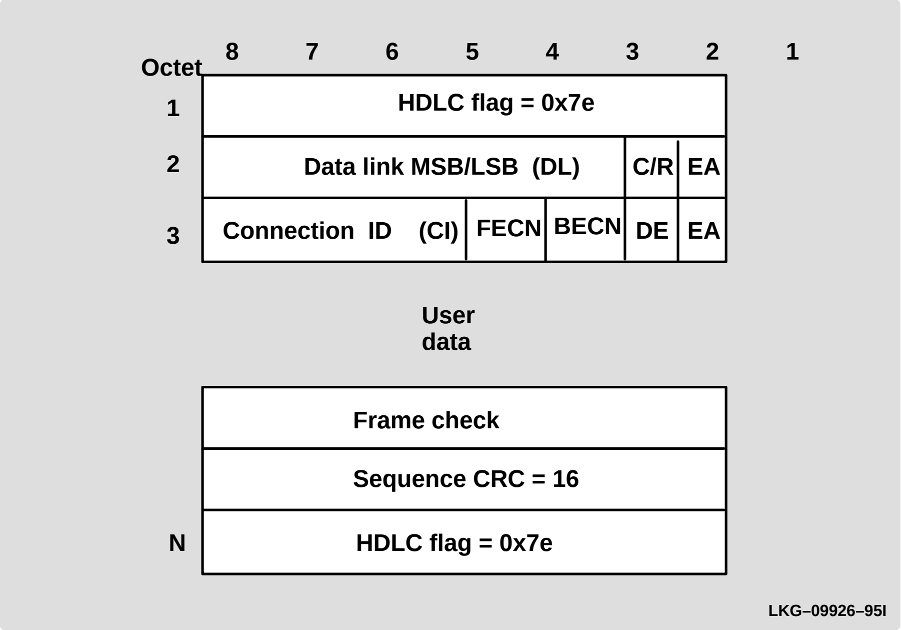


Frame Relay Frame

An FR frame consists of a fixed size control field with variable sized encapsulated user data.

Figure 11–4 illustrates an LAPD frame format.

Figure 11–4 LAPD Frame Format



HDLC Flags

Located in the first and last octet, these flags indicate the beginning and end of the frame.

Data Link Connection Identifier (DLCI)

This 10-bit routing ID resides in bits 3-8 of octet 2 and bits 5-8 of octet three. The DLCI is the MAC address of the circuit. The DLCI allows the user and network management to identify the frame as being from a particular PVC. The DLCI enables multiplexing of several PVCs over one physical link.

Command/Response (C/R)

This is LAPD-specific and is not used by this version of FR.

Extended Address

This version of FR does not support extended addressing.

Forward Explicit Congestion Notification (FECN)

When this bit is set to 1, the FR backbone network notifies the user receiving the frames that congestion is occurring in the direction the frame being sent.

Backward Explicit Congestion Notification (BECN)

When this bit is set to 1, the FR backbone network notifies the user sending the frames that congestion is occurring in the opposite direction. The router then initiates a *throttle down* to the user-defined Committed Information Rate (CIR). The CIR for a PVC is supplied by the FR service provider and is configured using the **add permanent** command.

Discard Eligibility (DE)

The network may discard transmitted data exceeding the CIR on a PVC. The DE bit is set by the network end-node to indicate discard eligibility. This version of FR does not set the DE bit, although it does log the exception.

User Data

This field contains the protocol packet being transmitted. This field can contain a maximum of 8189 octets; however, the frame check sequence (FCS) can effectively detect errors on a maximum of 4096 octets of data.

Frame Check Sequence (FCS)

This field is the standard the 16-bit cyclic redundancy check (CRC) that HDLC and LAPD frames use. This field detects bit errors occurring in the bits of the frame between the opening flag and FCS.

Frame Forwarding Over the Frame Relay Network

When the FR protocol receives a packet for encapsulation, it compares the packet's network address to the entries in the ARP cache. If the ARP cache contains the DLCI number that matches the network address, the FR protocol encapsulates that packet into a frame and transmits the frame over its specified local DLCI. If the ARP cache does not contain a match, the FR protocol sends out an ARP request over all configured PVCs on the interface. When the appropriate end-point responds with an ARP response, the FR protocol adds its local DLCI that received the ARP response to the ARP cache. Subsequent data packets directed to the same network address are then encapsulated into a frame and sent out over its local DLCI.

Protocol Addresses

Protocol addresses can be mapped to FR network PVC addresses either statically or dynamically through ARP. Either method is protocol-dependent as illustrated in Table 11–1.

Note: Static protocol addresses are also referred to as static ARP entries. A static ARP entry is added to the configuration with the **add protocol-address** command.

Table 11–1 Protocol Address Mapping

Protocol Type	ARP Usage	Static Mapping	PVC Configured at Protocol Configuration
IP	Yes	Yes	No
IPX	Yes	Yes	No
DNA IV	Yes	Yes	No
OSI*	No	No	Yes

* You must configure OSI at the protocol level to map the protocol address to the FR PVC.

Multicast Emulation

Multicast emulation is an optional feature that allows protocols requiring multicast such as ARP to function properly over the FR interface. With multicast emulation, a frame is transmitted on each active PVC. By using the **enable** and **disable multicast** commands, you can turn this feature on or off. Protocols that utilize multicast are IP, IPX, DNA IV, and ARP.

Frame Relay Network Management

The supplier of the FR network backbone provides FR network management. It is management's responsibility to provide FR end stations (router) with status and configuration information concerning PVCs available at the physical interface.

The FR protocol supports both the ANSI Annex D management and the Interim LMI management entities. You can turn these entities on or off using the **enable** and **disable** configuration commands. Specifically, FR network management provides the following information:

- Notifies the FR end stations of additional PVCs (orphans) and whether they are active or inactive, or of any PVC deletions.

- Asynchronous PVC status notification separate from a router's polled status request.

- Flow control notification through the FECN and BECN bit settings.

- Notification of the availability of a configured PVC. The availability of a PVC is indirectly related to the successful participation of the PVC end-point in the *heartbeat polling* process that is detailed in the section, "Link Integrity Verification Report."

- Verifying the integrity of the physical link between the end station and network by using a *keep alive* sequence number interchange.

- Including CIR as part of the PVC status informational elements.

Though the FR interface supports both types of network management, it is not necessary for management to run on the FR backbone for the interface to operate over the FR backbone. For example, you may want to disable management for back-to-back testing.

Management Status Reporting

Upon request, FR management provides two types of status reports: a full status report and a link integrity verification report. A full status report provides information about all PVCs the interface knows about. A link integrity verification report verifies the connection between a specific end station and network switch. All status enquiries and responses are sent over DLCI 0 for ANSI Annex D or DLCI 1023 for interim LMI management.

Full Status Report

When the FR interface requires a full status report, the FR interface on the router sends a status enquiry message to management requesting a full status report. A status enquiry message is a request for the status of all PVCs on the interface. Upon receiving this request, FR management must respond with a full status report consisting of link integrity verification element and a PVC status information element for each PVC.

The PVC status information element contains the local DLCI number for that PVC, whether the PVC is active or inactive, and whether the PVC is new or is an existing PVC that management already knows about. The link integrity verification element is discussed in the next section.

Note: The number of PVCs supplied at the FR interface is restricted by the network frame size and the amount of individual PVC information elements that can fit into a full status report. For example, 202 is the maximum number of PVCs for a network with a 1K frame size.

Link Integrity Verification Report

The link integrity verification report, sometimes referred to as *heartbeat polling*, contains the link integrity verification element. This element is where the exchange of the send and receive sequence numbers takes place. By exchanging sequence numbers, management and the end station can evaluate the integrity of the synchronous link. The send sequence number is the current send sequence number of the message originator. The receiver looks at this number and compares it to the last send sequence number to verify that this number is incrementally correct. The receive sequence number is the last send sequence number that the originator sent out over the interface. It is the receiver's responsibility to place a copy of the send sequence number into the receive sequence number field. This way the originator can ensure that the receiver receives and interprets the frames correctly.

When an end station fails to participate in this polling process, all remote end stations with logically attached PVCs are notified through management's full status report mechanism.

Circuit Congestion

Circuit congestion occurs because the sender is transmitting faster than the allowable throughput, the receiver is too slow when processing the frames, or an intermediate backbone link is congested resulting in the sender transmitting faster than the resulting throughput. When this happens, the network must either drop packets, shutdown, or both. If the router is the sender of the frames, it implements a *throttle down* during network congestion, slowing packet transmission below the line speed approaching the CIR.

Committed Information Rate (CIR)

Any PVC that is added or learned is provided (by the FR service provider) a CIR. The CIR is a portion of the total throughput for the physical link between 300 bits per second (bps) and 1.54 Mbps (where 64K bps or a single DS0 channel is most common. The CIR is the data rate that the network commits to supporting. The CIR is defined with the **add permanent** command.

Orphan Circuit CIR

When an orphan circuit is learned, the router assigns it a CIR equal to the line access rate. During times of congestion, the CIR may be greater than those PVCs that were added during network configuration. If you are relying on the orphan to route important data, it is recommended that you add a PVC in place of the orphan. This way you can assign a CIR that is supported by the network.

CIR Monitoring and Adjustment

CIR monitoring and adjustment is an optional feature that ensures a data transmission burst rate above the configured CIR. This burst rate is configured using the **set cir-adjustment** command that establishes a value between 1 and 100. This value is multiplied by the configured CIR, which was set with the **add permanent-circuit** command, to establish the allowable burst rate for that circuit. For example, the CIR monitor adjust value is set to 4. This allows a burst rates up to 4 times the configured CIR. When using this feature, set the CIR adjustment value that prevents transmissions above the excess burst rate provided by the FR vendor. This feature is enabled using the **enable CIR-monitor** command.

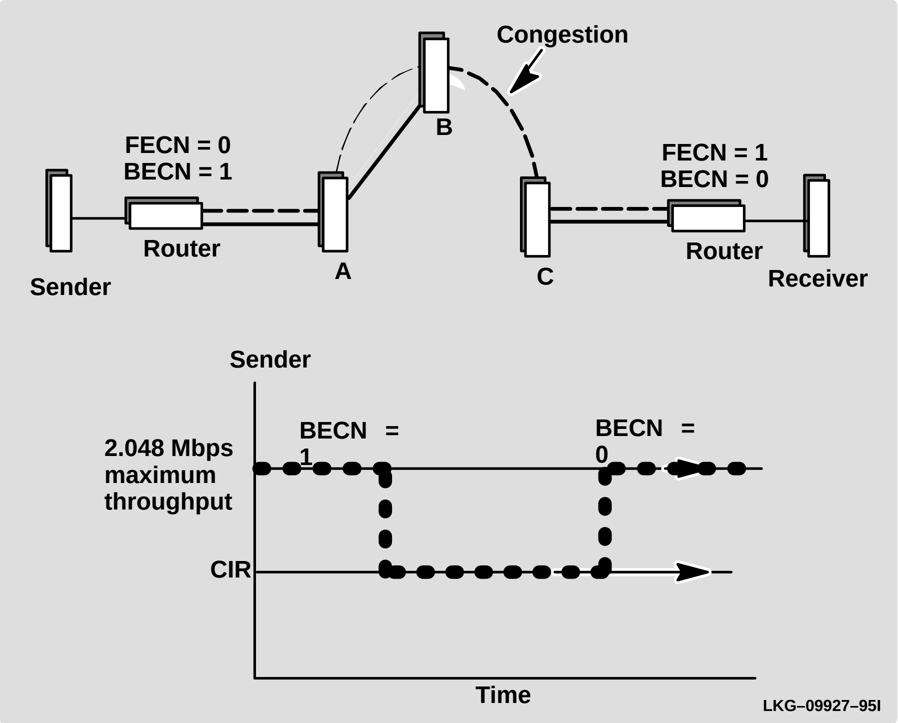
Congestion Notification and Avoidance

When congestion occurs, management is responsible for notifying the sender and receiver by sending out a FECN or a BECN signal. FECN and BECN are bits that are set in the frame to notify the receiver and sender, respectively, that congestion is occurring.

The example in Figure 11–5 shows a congestion condition at switch B. Management notifies the downstream node (switch C) and the end station (router) that congestion is occurring by setting the FECN bit on all outgoing frames. Management must also notify switch A and the other end station that congestion is occurring by setting the BECN bit. When the router receives the BECN, it is the router’s responsibility to throttle down the PVC’s throughput to the CIR. Once the BECN is cleared, the router resumes transmitting at the maximum throughput.

Figure 11–5 illustrates congestion notification and throttle down.

Figure 11–5 Congestion Notification and Throttle Down



Note: If multiple DLCIs are configured between two end stations when congestion occurs, it is possible that the other DLCI may be used to transmit data at a higher throughput until the congestion condition on the other DLCI is corrected.

12

The Point-to-Point Protocol

This chapter provides reference information about the Point-to-Point Protocol.

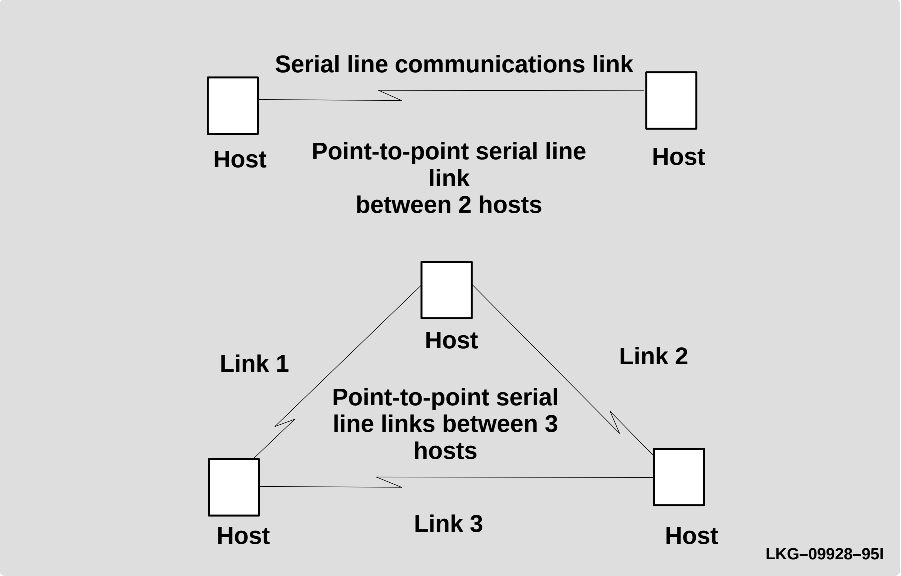
Point-to-Point Protocol (PPP) Overview

The Internet has grown considerably in the past few years with more and more hosts supporting TCP/IP. Some groups of hosts are connected to Local Area Networks (LANs) supporting different technologies (for example, Ethernet and token ring), while others connect to Wide Area Networks (WANs) such as an X.25 Public Data Network.

A large number of hosts are connected through one of the oldest methods of data communications – the point-to-point link. This type of connection can be described as a channel or link that has only two terminals.

Figure 12–1 shows some examples of point-to-point serial links.

Figure 12–1 Examples of Point-to-Point Links


Serial line communications link
Host
Point-to-point serial line link between 2 hosts
Host
Host
Link 1
Link 2
Point-to-point serial line links between 3 hosts
Host
Link 3
Host
LKG–09928–95I

PPP provides a method for transmitting protocol datagrams at the Data Link Layer over serial point-to-point links. PPP currently supports synchronous data transmission. PPP provides the following services:

- **Link Control Protocol (LCP)** – To establish, configure, and test the link connection.
- **Encapsulation protocol** – To encapsulate protocol datagrams (through the HDLC method) over serial point-to-point links.
- **Network Control Protocols (NCP)** – To establish and configure different network-layer protocols. PPP allows the use of multiple network layer protocols.

Note: PPP currently supports the AppleTalk Control Protocol (ATCP), the Bridging Network Control Protocol (BNCP), the DECnet Protocol Control Protocol (DNCP), the Internet Protocol Control Protocol (IPCP), the IPX Control Protocol (IPXCP), and the OSI Control Protocol (OSICP).

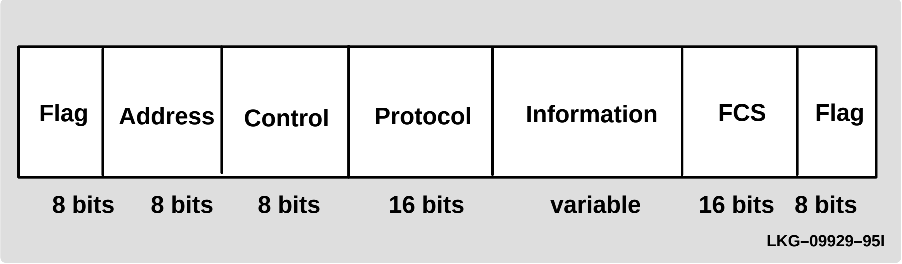
To establish data transmission over a point-to-point link, the originating PPP first sends LCP packets to configure and test the data link. After the link is established, PPP sends NCP packets to choose and configure one or more network layer protocols. After network layer protocols are configured, datagrams from each network layer can be sent over the link. The next sections explain these concepts in more detail.

PPP Data Link Layer Frame Structure

The Point-to-Point Protocol transmits data frames that have the same structure as High-level Data Link Control (HDLC) frames. PPP uses a bit-oriented, synchronous transmission method with a single-frame format for all data and control exchanges.

Figure 12–2 illustrates the PPP frame structure and is followed by a detailed description of each frame.

Figure 12–2 PPP Frame Structure



Flag Fields

The flag field begins and ends each frame with a unique pattern of 01111110. Generally a single flag ends one frame and begins the next. All active stations attached to the link continuously search for the flag sequence to synchronize the start of the next frame. The sending and receiving stations use a process called bit stuffing to avoid the loss of synchronization due to the arbitrary appearance of the 01111110 bit pattern within the data stream.

When bit stuffing, the sending station adds a 0 after the fifth 1 in a non-flag bit stream. Then the receiving station monitors the bit stream. If the receiver detects a bit pattern of five ones, it examines the sixth bit. If the sixth bit is a 0, the receiver deletes it. If the sixth bit is a 1 and the seventh bit is a 0, the receiver accepts the combination as the end of frame flag.

Address Field

The address field is a single octet (8 bits) and contains the binary sequence 11111111 (0xff hexadecimal). This is known as the All-Station Address. PPP does not assign individual station addresses.

Control Field

The control field is a single octet and contains the binary sequence 00000011 (0x03 hexadecimal). This sequence identifies the Unnumbered Information (UI) command with the P/F bit set to zero.

Protocol Field

The protocol field is defined by PPP and is not found in the true HDLC frame format. The protocol field is 2 octets (16 bits) and its value identifies the protocol datagram encapsulated in the Information field of the frame.

Protocol field values in the CXXX range indicate that datagrams belong to the Link Control Protocol (LCP). Values in the 8XXX range indicate that datagrams belong to the Network Control Protocols (NCP). Values in the 0XXX range identify the network protocol of specific datagrams.

Information Field

The information field contains the datagram for the protocol specified in the protocol field. This is zero or more octets. This information field is present for all I-frames and only some U-frames.

When the protocol type is LCP, exactly one LCP packet is encapsulated in the information field of PPP Data Link Layer frames.

Frame Check Sequence (FCS) Field

The frame check sequence field is a 16-bit cyclic redundancy check (CRC).

The PPP Link Control Protocol (LCP)

PPP's Link Control Protocol (LCP) establishes, configures, maintains, and terminates the point-to-point link. This process is carried out in four phases:

1. Before exchanging any IP datagram, LCP first opens the connection through an exchange of Configure packets. After this exchange is complete (with the reception of a Configure-Ack packet), the link enters the OPEN state. LCP handles only configuration parameters associated with the link; it does not handle configuration of any of the network-layer protocols.

2. After the link enters the OPEN state, LCP tests to determine if the quality of the link is sufficient to bring up the network protocols.

3. Once LCP has determined that the quality of the link is sufficient to bring up network layer protocols, the appropriate NCP configures the network protocols and brings them up and takes them down. If LCP closes the link, the network layer protocols are first notified.

4. Finally, LCP has the ability to terminate the link at any time. This is usually done at the request of the user but may occur because of the loss of a carrier or the expiration of an idle-period timer.

LCP Packets

Before exchanging any IP datagrams, LCP first opens the connection through an exchange of Configure packets. There are three classes of LCP packets:

- **Link Establishment Packets** – Establishes and configures a point-to-point link.

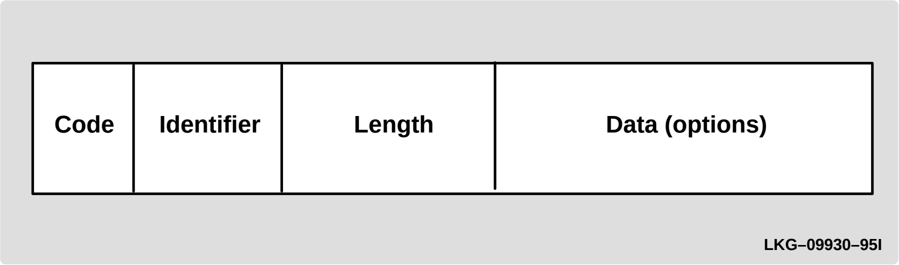
- **Link Termination Packets** – Terminates a link.

- **Link Maintenance Packets** – Manages and debugs a link.

Only one LCP packet is encapsulated in the information field of PPP Data Link Layer frames. In the case of LCP packets, the protocol field reads “Link Control Protocol” (C021 hexadecimal).

Figure 12–3 illustrates the structure of the LCP packet and is followed by a detailed description of each field.

Figure 12–3 PPP Frame Structure



Code

The code field is one octet in length and identifies the type of LCP packet. The codes in Table 12–1 distinguish the packet types. They are described in more detail in later sections.

Table 12–1 LCP Packet Codes

Code	Packet Type
1	Configure-Request (Link Establishment)
2	Configure-Ack (Link Establishment)
3	Configure-Nak (Link Establishment)
4	Configure-Reject (Link Establishment)
5	Terminate-Request (Link Termination)
6	Terminate-Ack (Link Termination)
7	Code-Reject (Link Termination)
8	Protocol-Reject (Link Maintenance)
9	Echo-Request (Link Maintenance)
10	Echo-Reply (Link Maintenance)
11	Discard-Request (Link Maintenance)

Identifier

The identifier field is one octet in length and is used to match packet requests and replies.

Length

The length field is two octets in length and indicates the total length (including all fields) of the LCP packet.

Data (Option)

The data field is zero or more octets as indicated by the length field. The format of this field is determined by the code.

Link Establishment Packets

Link Establishment Packets establish and configure a point-to-point link including the following packet types:

- **Configure-Request** – LCP packet code field is set to 1. LPC transmits this packet type when you want to open a point-to-point link. Upon receiving a Configure-Request, a peer station's LCP entity must send an appropriate reply.

- **Configure-Ack** – LCP packet code field is set to 2. The peer transmits this packet type when every configuration option in a Configure-Request packet is acceptable. Upon receiving the Configure-Ack (ack = acknowledgement), the originating station checks the Identifier field. This field must match the one from the last transmitted Configure-Request or the packet is invalid.

- **Configure-Nak** – LCP packet code field is set to 3. The peer transmits this packet type when some part of the configuration option in a Configure-Request packet is unacceptable. The Identifier field is copied from the received Configure-Request and the Data (option) field is filled with the received unacceptable configuration options. The Identifier field must match the one from the last transmitted Configure-Request or the packet is invalid and is discarded.

When the originator receives a Configure-Reject packet, a new Configure-Request packet is sent that includes modified, acceptable configuration options.

- **Configure-Reject** – LCP packet code field is set to 4. The peer transmits this packet type when some part of the configuration options in a Configure-Request packet is unacceptable. The Identifier field is copied from the received Configure-Request and the Data (option) field is filled with the received unacceptable configuration options. The Identifier field must match the one from the last transmitted Configure-Request or the packet is invalid and is discarded.

  When the originator receives a Configure-Reject packet, a new Configure-Request packet is sent that does not include any of the configuration options received in the Configure-Reject packet.

Link Termination Packets

Link Termination Packets terminate a link and include the following packet types:

- **Terminate-Request** – LCP packet code field is set to 5. LCP transmits this packet type when a point-to-point link needs to be closed. These packets are sent until a Terminate-Ack packet is sent back.

- **Terminate-Ack** – LCP packet code field is set to 6. Upon receiving a Terminate-Request packet, this packet type must be transmitted with the code field set to 6. Reception of a Terminate-Ack packet that was not expected indicates that the link is closed.

Link Maintenance Packets

Link Maintenance Packets manage and debug a link, and include the following packet types:

- **Code-Reject** – LCP packet code field is set to 7. The transmission of this packet type indicates that one of the communicating LCP entities is faulty or incomplete. This error must be reported back to the sender of the unknown code by transmitting an LCP packet with the code field set to 7. This situation ends in the closing of the link.

- **Protocol-Reject** – LCP packet code field is set to 8. The transmission of this packet type indicates that a PPP frame that was received contains an unsupported or unknown protocol. Upon receiving a Protocol-Reject packet, the peer stops transmitting the incorrect protocol.

- **Echo-Request *and* Echo-Reply** – LCP packet code fields are set to 9 and 10 respectively. LCP transmits these packet types in order to provide a Data Link Layer loopback mechanism for both directions on the link. This feature is useful in debugging, for example, a faulty link to determining link quality. These packets are sent only when the link is in the Open state.

- **Discard-Request** – LCP packet code field is set to 11. LCP transmits this packet type to provide a data *sink* for Data link Layer testing. A peer that receives a Discard-Request MUST throw away the packet. This is useful in debugging a link. These packets are sent only when the link is in the Open state.

The PPP Network Control Protocols

PPP has a family of Network Control Protocols (NCPs) for establishing and configuring different network-layer protocols. The NCPs are responsible for configuring, enabling, and disabling the network-layer protocols on both ends of the point-to-point link. NCP packets cannot be exchanged until LCP opens the connection and the link reaches the OPEN state.

PPP currently supports the following Network Control Protocols:

- AppleTalk Control Protocol (ATCP)

- Bridging Network Control Protocol (BNCP)

- DECnet Control Protocol (DNCP)

- IP Control Protocol (IPCP)

- IPX Control Protocol (IPXCP)

- OSI Control Protocol (OSICP)

AppleTalk Control Protocol

ATCP is specified in Request for Comments (RFC) 1378. Digital's implementation of ATCP supports the AppleTalk-Address option.

To transmit an AppleTalk datagram over PPP, the AppleTalk packet is encapsulated in the Information field of the PPP data frame. The encapsulated AppleTalk packet begins with an extended Datagram Delivery Protocol (DDP) header.

There are two modes for AppleTalk over PPP, full router and half router. In full-router mode, the point-to-point network is visible to other AppleTalk routers. In half-router mode, the point-to-point network is invisible to other routers, but it still transmits AppleTalk routing information and data packets.

To set up your network for full-router mode, each router on the PPP link needs to have a common network number, a common zone name, and a unique node number. If you configure one end of the PPP link with a non-zero network number, you must also configure that end to have a non-zero node number and to have a zone name. In this case, the other end of the link must have either:

- The same network number and zone name and a different node number.

  OR

- Network and node numbers set to zero. The router learns network and node numbers from the configured router.

To set up your network for half-router mode, configure both routers on the PPP link so that network and node numbers are set to zero and no zone name is used.

Bridging Network Control Protocol

BNCP is specified in RFC 1220. Digital's implementation of BNCP supports IEEE 802.5 Line Identification Option and the Tinygram Compression Option.

DECnet Control Protocol

DNCP is specified in RFC 1376. Digital's implementation does not support any DNCP options.

IP Control Protocol

IPCP is specified in RFC 1332. Digital's implementation supports all IPCP options. They are:

- IP Addresses
- IP Compression Protocol
- IP Address

IPX Control Protocol

IPXCP is specified in RFC 1552. Digital's implementation does not support any IPXCP options.

Note: For more information on IPXCP, see Chapter 10, "The IPX Protocol," in this supplement.

OSI Control Protocol

OSICP is specified in RFC 1377. Digital's implementation of OSICP does not support any options.

13

Bandwidth Reservation and Priority Queuing

This chapter explains the bandwidth reservation and priority queuing features currently available for serial interfaces.

Bandwidth Reservation

Bandwidth reservation is an algorithm for deciding which packets to drop when demand (traffic) exceeds supply (throughput). Bandwidth reservation is not used until more than 100% of the available line bandwidth is requested.

Bandwidth reservation guarantees (reserves) transmission bandwidth for a network connection. This reservation feature allocates minimum percentages of total connection bandwidth for specified classes of traffic.

These reserved percentages are a guaranteed minimum slice of bandwidth for the network connection. If a network is operating to capacity, any one message can only be transmitted until it uses the required bandwidth allocated for its class. In this case, the transmission is held until other bandwidth transmissions are satisfied. In the case of a light traffic path, a packet stream can use bandwidth exceeding its allowed minimum (up to 100%) if there is no other traffic.

Bandwidth reservation is really a safeguard. In general, a network does not attempt to use greater than 100% of its line speed. If so, a faster line is probably needed. The “bursty” nature of traffic, however, can drive the requested transmission rate to be greater than 100% for a short time. In these cases, bandwidth reservation is enabled and the more important traffic is ensured delivery (is not discarded).

Priority Queuing

Bandwidth reservation allocates percentages of total connection bandwidth for specified traffic classes. With priority queuing, each bandwidth class can be assigned one of the following priority level settings:

- Urgent
- High
- Normal (the default setting)
- Low

All traffic set to Urgent is sent first within the class. These are followed by High, Normal, and then Low messages respectively. When all Urgent packets have been transmitted, High packets are transmitted until complete (or until new Urgent priority messages are queued). Only when there are no Urgent, High, or Normal packets remaining are the Low priority packets transmitted.

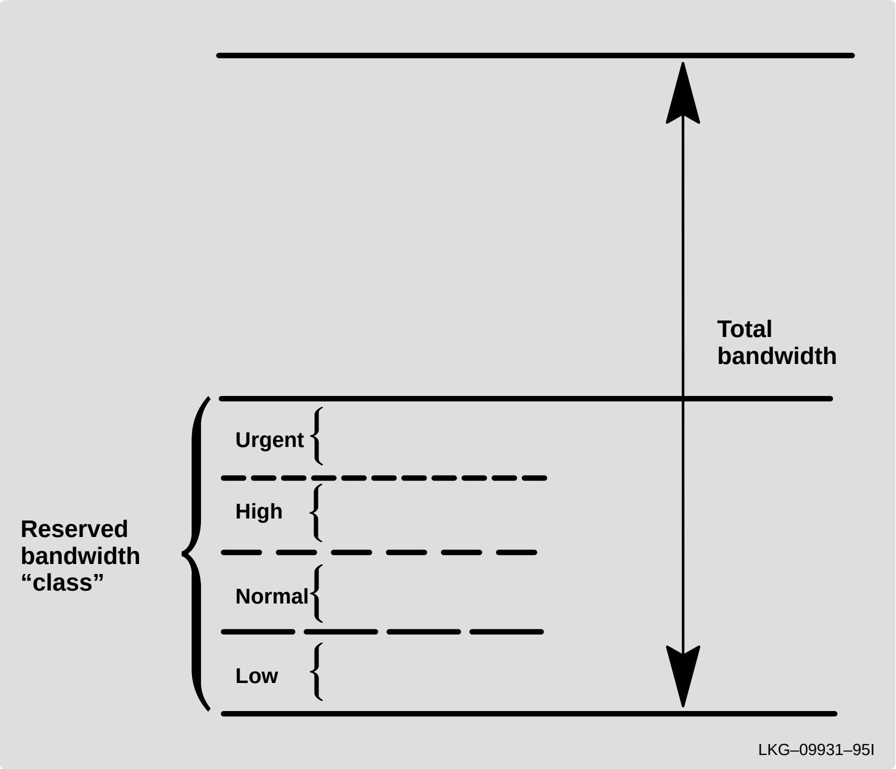
The priority settings in the bandwidth class have no effect on other bandwidth classes. No one bandwidth class has priority over the others.   Currently, you can only map a network protocol (or several grouped protocols) or filters to a class. The default priority is Normal if no priority settings are assigned.

Bandwidth Reservation with Priority Queuing

With just priority queuing, only the highest priority traffic is guaranteed delivery. In cases of heavy high priority traffic, lower priority levels can be overlooked. By combining priority queuing with bandwidth reservation, packet transmission can be allocated to all bandwidths.

Figure 13–1 illustrates bandwidth reservation working with priority queuing.

Figure 13–1 Bandwidth Reservation with Priority Queuing



Filtering and Bandwidth Reservation

Previously, you could only map a network protocol or several grouped protocols to a class. Using bandwidth reservation, you can assign filters (through the **assign** command) to very specific types of traffic. The following filters can be assigned:

- IP tunneling
- SDLC tunneling over IP
- Rlogin

- Telnet
- SNA
- SNMP
- Multicast
- DLSw
- MAC Address

Filters and Tags For Multicast Addressing and Mac Addressing

A filter is available to filter IP multicast traffic. This filter appears in the normal filter list.

MAC Address filtering is handled by a joint effort between bandwidth reservation and the MAC Filtering Feature (MCF) using tags. For example, a user with bandwidth reservation is able to categorize bridge traffic by assigning a tag to it.

The tagging process is done by creating a filter item in the MAC Filtering configuration console and then assigning a tag to it. This tag is then used to set up a bandwidth class for all packets associated with this tag. Tag values must currently be in the range of 1-64.

Note: The current software release supports applying tags to ONLY bridged packets, and allows ONLY the MAC Address fields of the packet to be used in applying the tag. Up to five tagged MAC filters can be set from 1 to 5. TAG1 is searched for first, then TAG2, and so on up to TAG5. A single MAC filter tag can consist of any number of MAC Addresses set in MCF.

Once a tagged filter is created in the MAC Filtering configuration process, it is then assigned a class and priority in the bandwidth reservation configuration process. The **tag** command is then used in the bandwidth reservation process to reference the tag.

Tags can also refer to “groups,” as in the example of the IP Tunnel. IP Tunnel endpoints can belong to any number of groups. Packets are assigned to a particular group through the tagging feature of MAC Address filtering.

Order Of Precedence

It is possible for a packet to fall under several filterable classes. For example, an IP Tunneled bridged packet for SNA with a filter for a MAC Address. The order for resolving this packet is the following:

1. MAC Address match for bridging (IP/ASRT) tag 1 to tag 5
2. NETBIOS for bridging (IP/ASRT)
3. SNA for bridging (IP/ASRT)
4. IP tunneling (IP)
5. SDLC relay (IP)
6. Multicast (IP)
7. SNMP (IP)
8. Rlogin (IP)
9. Telnet (IP)
10. DLSw(IP)

14

The DLSw Protocol

This chapter describes router implementation of the Data Link Switching (DLSw) protocol.

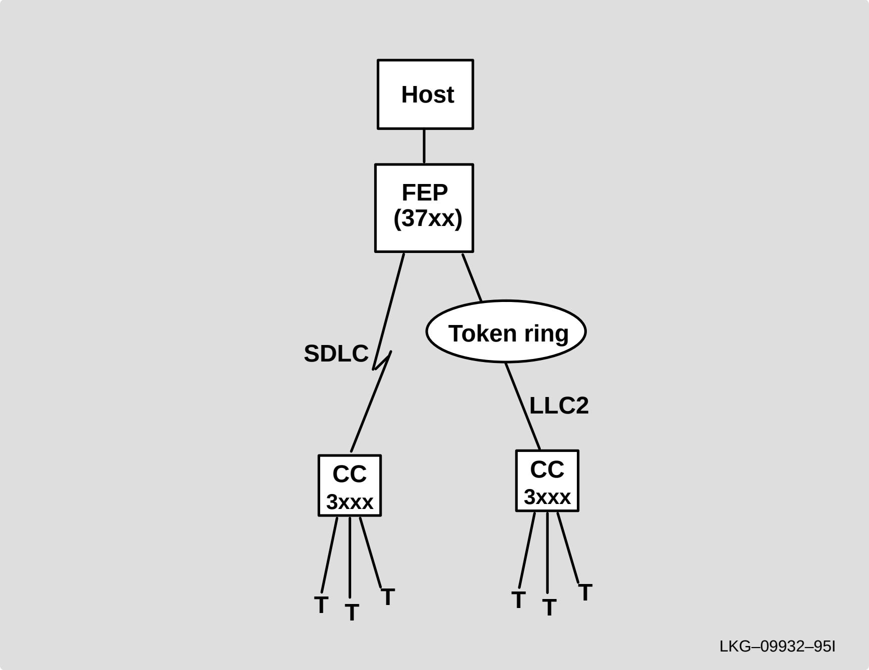
For DLSw configuring and monitoring procedures, refer to the chapters “Configuring DLSw” and “Monitoring DLSw” in the *Routing Protocols User’s Guide*.

Data Link Switching Overview

Digital’s Data Link Switching product offers a wide range of functionality designed to facilitate integration with IBM’s Systems Network Architecture (SNA) into a heterogeneous, multi-protocol network. DLSw is especially effective at eliminating SNA session time-outs and reducing Wide Area Network (WAN) overhead on shared circuits. In addition, DLSw incorporates OSPF-based, TCP/IP networking standards for communication supporting SNA sessions.

Figure 14–1 shows a basic SNA subarea network.

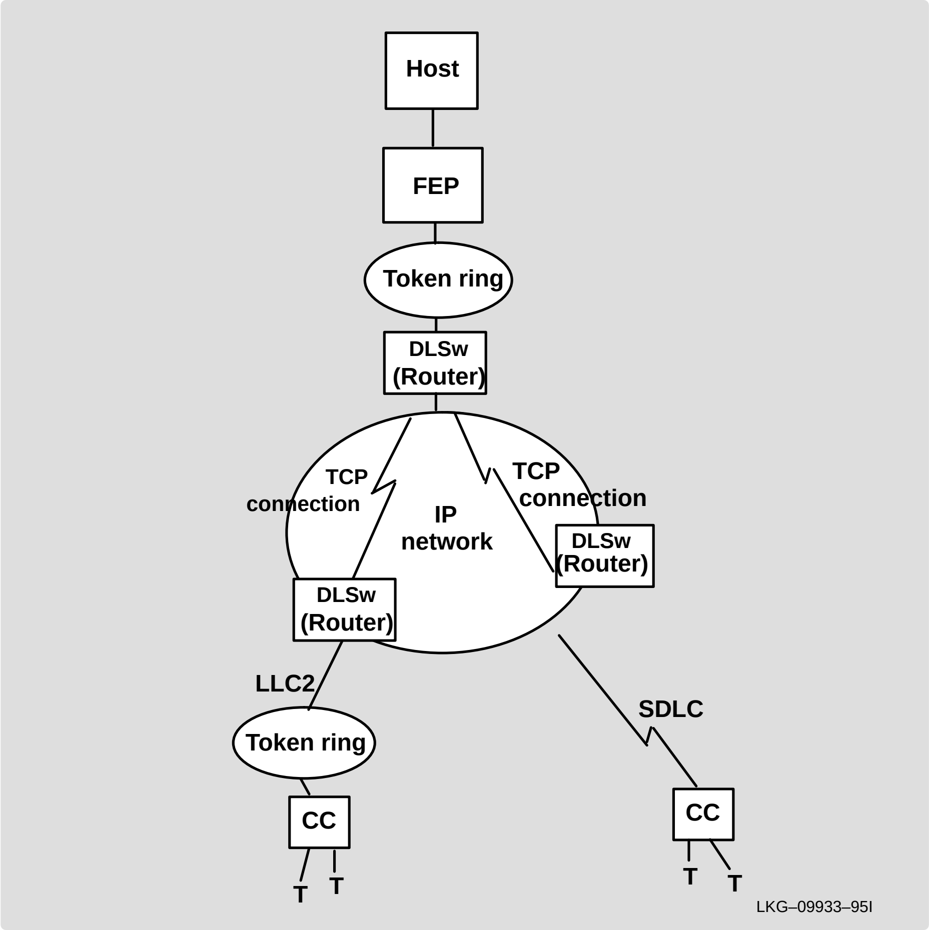
Figure 14–1 Basic SNA Network



It illustrates terminals connected to the Host/Front End Processor (FEP) through the Cluster Controller (CC) using Synchronous Data Link Control (SDLC) and IEEE token ring/Logical Link Control Type 2 (LLC2). The FEP (or Communications Controller) can be either a 3705, 3725, 3740, or a 3745. The supported Cluster Controllers are a 3274, 3174, or a compatible.

Figure 14–2 shows a network sharing WAN links in a SNA network configured with DLSw.

Figure 14–2 SNA Network with DLSw

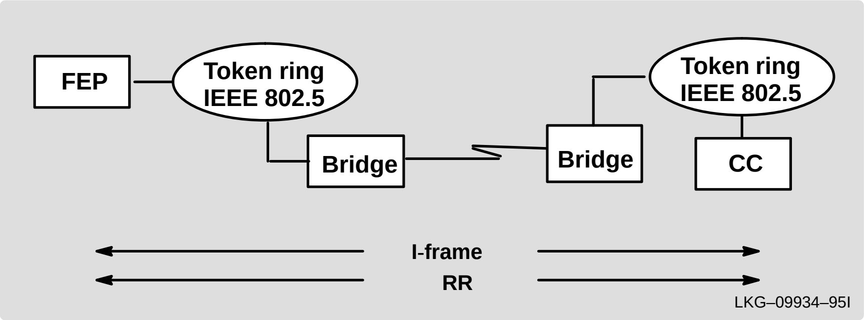


DLSw provides data link termination at the router. DLSw does this by terminating control at the SDLC or Logical Link Control Type 2 (LLC2) sessions at the router. Data is transported over the IP network through the TCP connection. LLC2 attached devices and SDLC attached devices, when connected through DLSw, appear as token ring attached devices to the FEP.

In bridged network configurations, data link sessions are established directly between the FEP and the Cluster Controller (CC). All traffic is forwarded through the FEP to the CC through bridges.

For example, in Figure 14–3, an Information Frame (I-Frame) is sent across the bridge by the FEP to the CC. The CC returns a Received Ready (RR) poll that is also sent across the bridge.

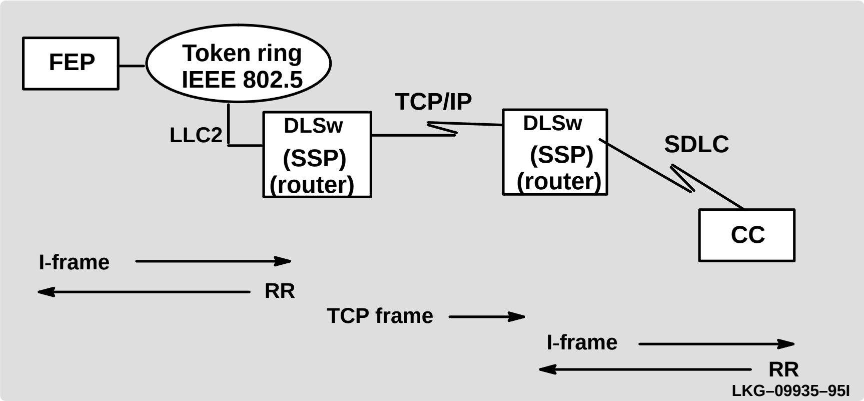
Figure 14–3 Bridging



Conversely, in DLSw, the data link is terminated at the router.

Figure 14–4 shows an example of an SDLC attached device appearing as IEEE 802.5 token ring attached to the FEP.

Figure 14–4 Data Link Switching (DLSw)



The user data is acknowledged locally at the router. DLSw uses a TCP connection to transfer user data. The advantage of the DLSw local acknowledgment is that there is less risk of link level time-outs.

Note: Only I-Frames are transmitted across the IP network. This allows for more reliable data transmissions because data is terminated at the end station routers.

DLSw uses the Switch-to-Switch Protocol (SSP) over TCP/IP between DLSw routers to:

- Initiate a connection
- Transfer user data
- Terminate a connection

When DLSw receives an I-Frame, it acknowledges it with a Receive-Ready (RR). The DLSw then transports the data frame through TCP to the DLSw peer. The recipient DLSw then transmits the data to the end station.

The following sections describe how LLC and SDLC connections and data flow are established in the SNA network.

LLC Connection within DLSw

This section provides an overview of an LLC connection within DLSw. It describes the following:

- DLSw LLC Connection Establishment
- DLSw LLC2 Connection Establishment and Data Exchange
- DLSw LLC2 Error Handling and Connection Termination

Note: LLC is available only with the DLSw protocol.

For an overview of a typical LLC connection establishment and the initial route discovery phase of the LLC connection, see the section “LLC Connection Establishment.”

DLSw locally terminates the LLC2 connection. This is transparent to the LLC entities within the end stations. The advantages of local termination are a reduction in WAN traffic due to the elimination of control traffic and reduction of connection outage due to session time-outs. The DLSw maintains a TCP connection between them over which these LLC2 connections are multiplexed (without losing their identities.) It is the responsibility of DLSw to deliver frames that it receives from the LLC2 connection partner (DLSw) using TCP.

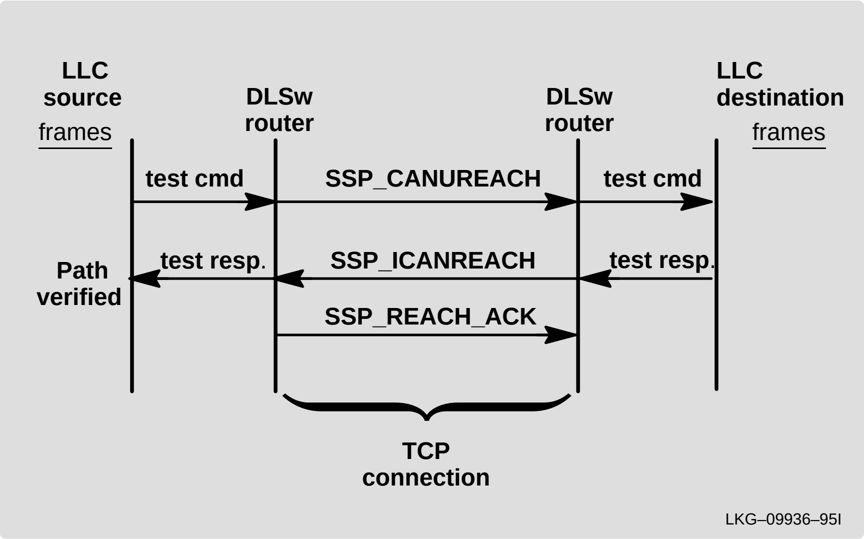
The following sections discuss DLSw peer discovery and connection and LLC2 flow control using SSP.

DLSw LLC Connection Establishment

When the LLC source peer sends out a TEST command frame to establish an LLC connection, DLSw terminates the frame at the local connection and sends out a SSP_CANUREACH frame to the remote DLSw device. When the remote DLSw receives the SSP_CANUREACH, it generates a TEST command to the remote LLC destination peer. The LLC destination device generates a TEST response frame that is terminated at the remote DLSw device. The remote DLSw device then generates and transmits a SSP_ICANREACH frame to the source DLSw peer. The DLSw source peer then generates a TEST response frame and transmits it back to the LLC sender (Figure 14–5).

Note: For Figure 14–5 through Figure 14–8, the DLSw transport on each end is the LLC.

Figure 14–5 DLSw LLC Connection Establishment



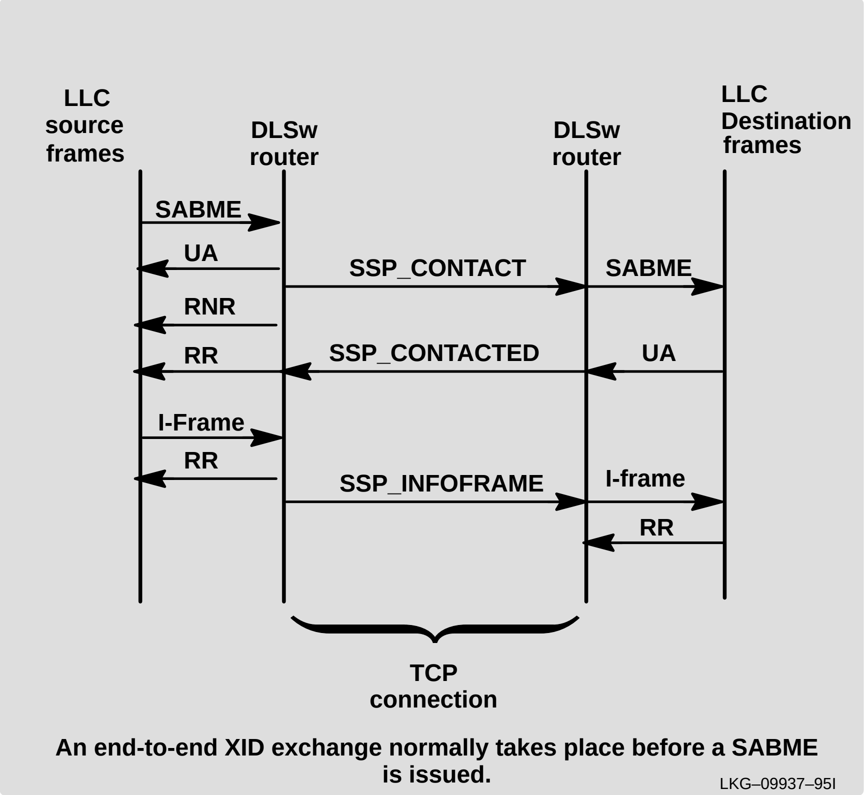
For details on SSP messages, refer to the document, *Data Link Switching: Switch-to-Switch Protocol (SSP) Network Working Group Request For Comments: 1434* available from IETF.

DLSw LLC2 Connection Establishment and Data Exchange

When the LLC2 establishes a connection, it relies on different timers to ensure the integrity of the connection as data passes between them. In large networks, connection establishment can be hindered by network delays coupled with short connection time-out values at the end stations. To reduce the occurrences of these time-outs, the bridging router uses the DLSw protocol to locally terminate the LLC2 connection, while maintaining reachability to the remote LLC peer through the TCP connection.

Figure 14–6 illustrates the DLSw connection establishment and data exchange.

Figure 14–6 LLC Connection Verification



DLSw LLC2 Error Handling and Connection Termination

The DLSw protocol is responsible for detecting LLC errors while processing a frame. When DLSw detects an error, either an REJ (Reject) or FRMR (Frame Reject) is sent back to the LLC peer. An FRMR frame requires a SABME to reset the connection or a DISC to terminate the connection. Refer to your LLC 802.2 specification for details on error conditions that elicit a FRMR response.

Figure 14–7 illustrates DLSw LLC error handling.

Figure 14–7 DLSw LLC Error Handling

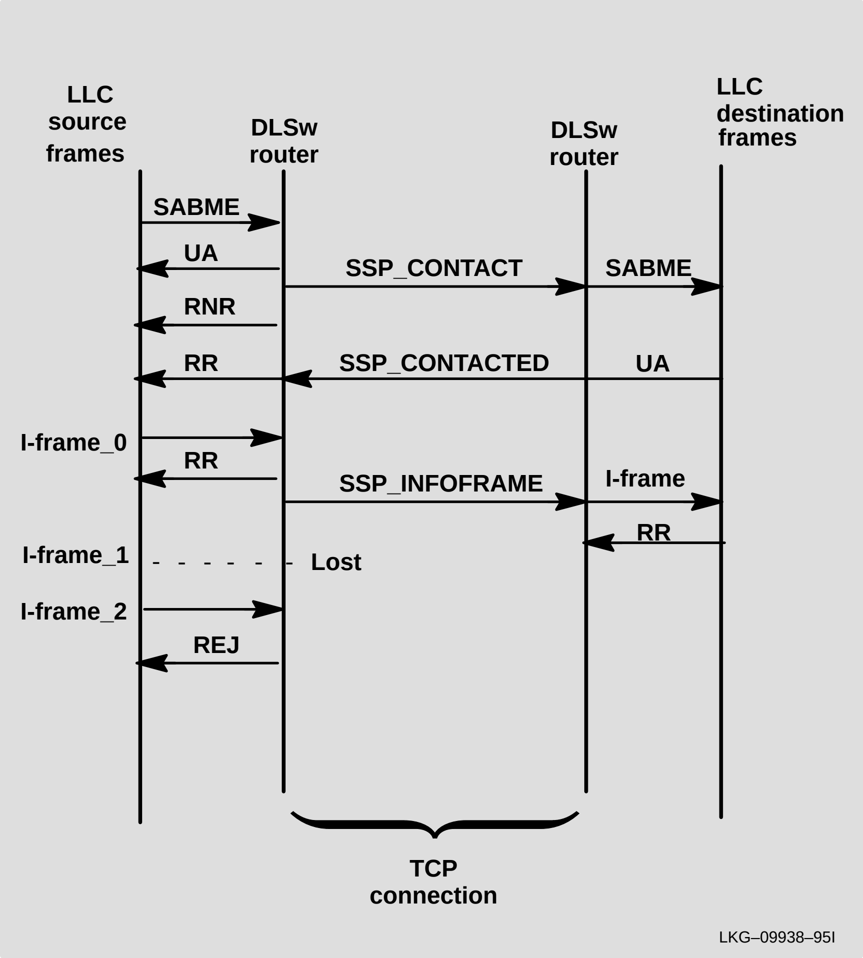
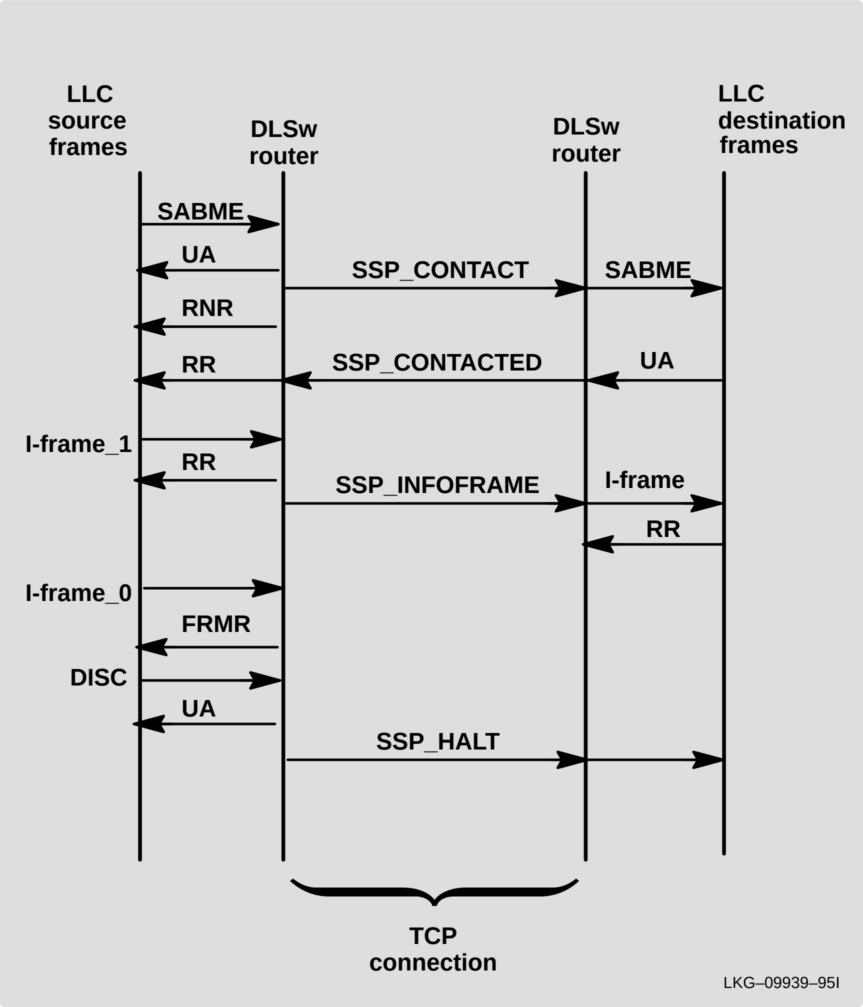


Figure 14–8 illustrates DLSw LLC connection termination.

Figure 14–8 DLSw LLC Connection Termination

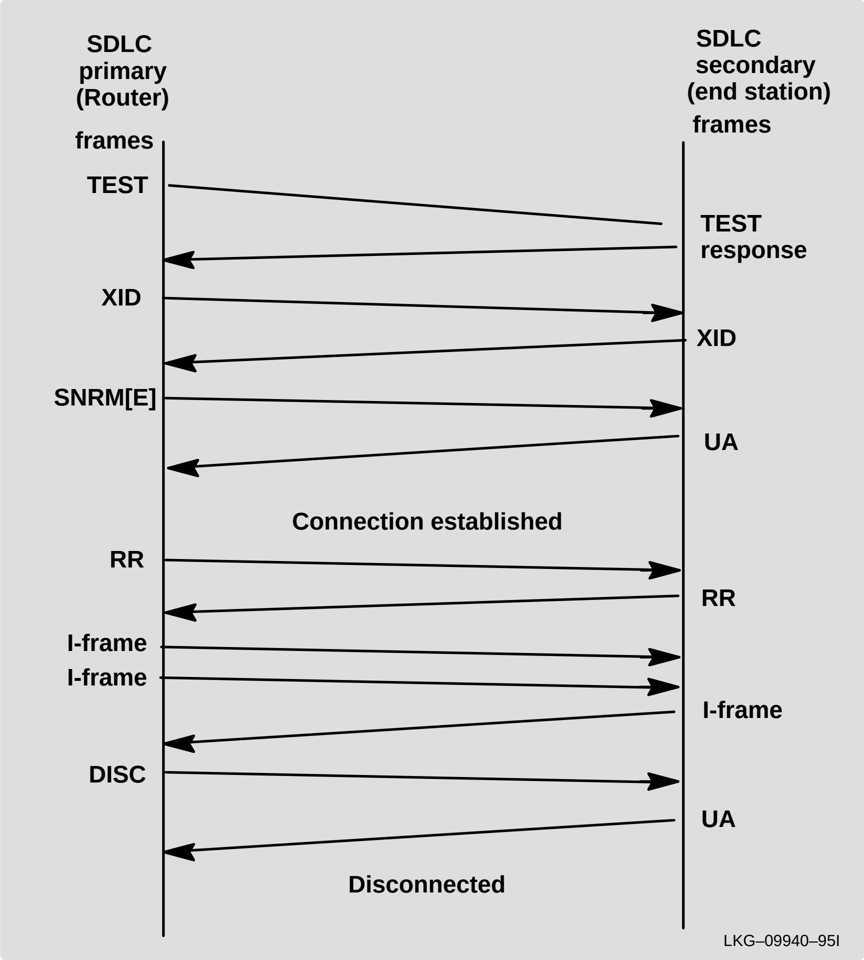


DLSw within SDLC Connection Establishment

The SDLC protocol defines the primary and secondary station roles. Only the primary station can establish a connection. The secondary station cannot transmit data unless polled. SDLC connects to DLSw after a Unnumbered Acknowledgement (UA) is received.

Figure 14–9 illustrates how DLSw within a SDLC connection is established.

Figure 14–9 DLSw within SDLC Connection Verification



DLSw Groups

To enable communication in DLS, normally a DLS router is configured with the IP addresses of the DLS routers that you want to communicate with (DLS neighbors). Because of the overhead of TCP connections and the fact that not all sites need to communicate to all other sites, the topology of DLS connectivity is a subset of a fully connected topology.

In addition to the static entry (through IP addresses) of DLS neighbors, DLS neighbors can dynamically find each other by using the DLS Group Membership functionality. DLS Groups alleviate the need for long lists of static IP addresses and the cost associated with maintaining them. A DLS router can be a member of up to 64 multiple groups.

There are two types of groups: Client/Server, in which a member of the group is designated as a client or a server, and Peer to Peer. In a Client/Server group, Server designated routers only form DLS connections with client-designated routers. In a Peer to Peer group, all routers form connections to each other.

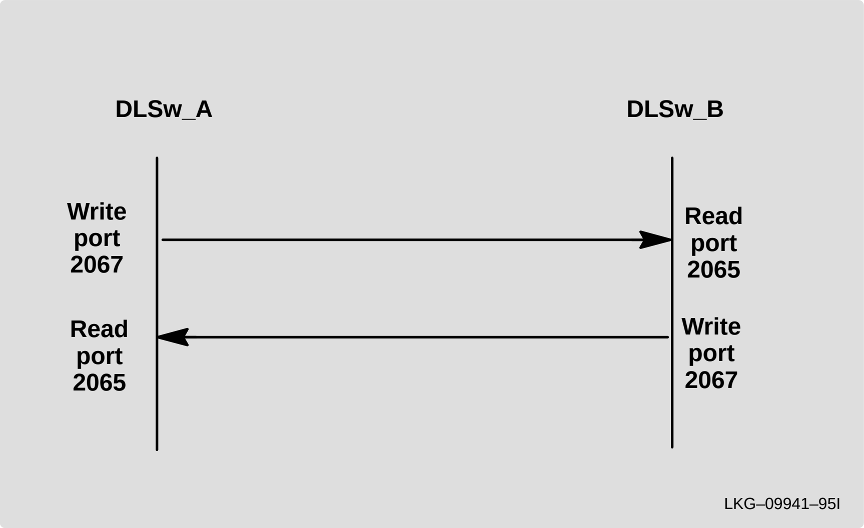
DLSw Peer Discovery and TCP Connection

This section describes DLSw peer and TCP connection. When the DLSw protocol is first initiated at start-up, it seeks out all other DLSw peers, either through a static configuration in SRAM (for example, it is manually entered into the configuration table), or through the OSPF multicast protocol. When contact is verified, two TCP connections (a read port and a write port) are opened to the other DLSw peer. The DLSw needs two connections because either side can initiate the DLSw connection at the same time.

The read port (2065) is a passive port. A passive port is one that only receives data but does not transmit (that is, a read port). When the DLSw peer sends out a frame, it actively seeks this port. The write port (2067) is an active port. When the DLSw sends out a frame, it sends it from this port.

Figure 14–10 shows DLSw read and write ports.

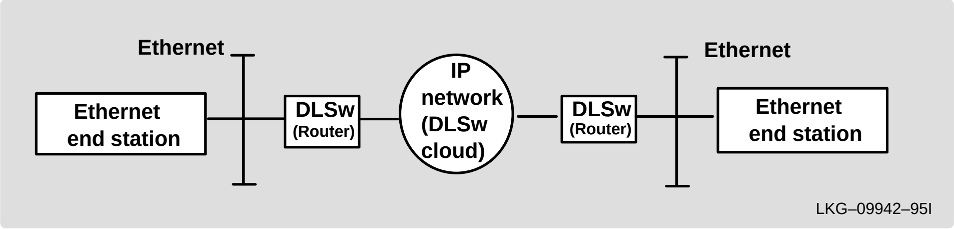
Figure 14–10 DLSw TCP Read and Write Ports



DLSw over Ethernet

DLSw services are provided for the Ethernet end stations. Figure 14–11 shows DLSw over Ethernet.

Figure 14–11 DLSw over Ethernet



To provide Ethernet access you need to:

1. Enable transparent bridging over respective Ethernet LANs as described in the *Bridging Configuration Guide*.

2. Configure bridging according to the ASRT configuration procedures described in the *Bridging Configuration Guide*. Bridging must be configured to allow the DLSw to receive SNA frames from the end stations. To enable switching of SNA traffic across DLSw cloud, you must specify the Service Access Point (SAP) and interface number as described in **open-sap** command. The interface number used is an Ethernet interface number.
3. Configure DLSw according to the procedures described in the “Configuring DLSw” chapter in the *Routing Protocols User’s Guide*.

There are two restrictions when DLSw is enabled in transparent bridge environment. They are as follows:

- Multiple DLSw routers cannot be used on the same extended LAN if they can access both the origin and target end stations. You can place more than one DLSw router at the origin LAN as long as they are not accessible to each other over the DLSw cloud. It is possible to use the MAC address filtering feature to overcome some of the topological restrictions.
- DLSw path cannot be in parallel with the transparent bridging path.

These restrictions occur because the spanning tree protocol which detects network loops is not executed across the DLSw protocol. Since the DLSw router is used only to switch SNA traffic, it is not a suitable candidate for spanning tree protocol participation.

DLSw over Token Ring

DLSw over token ring requires that suitable bridge behavior be enabled on LAN interfaces. You must also enable switching of SNA traffic across DLSw cloud specifying the SAP and interface number as described in **open-sap** command. The interface number used is a token ring interface number. There are two types of bridging capabilities that token ring LAN architecture supports:

- Source Routing Bridging (SRB)
- Transparent Bridging (STB)

Source Routing Bridging capability is available on all token rings on all platforms. However, there are limitations to the support of Transparent Bridging on token ring. STB requires special hardware support on the interface cards, called CAMs (Content Addressable Memory).

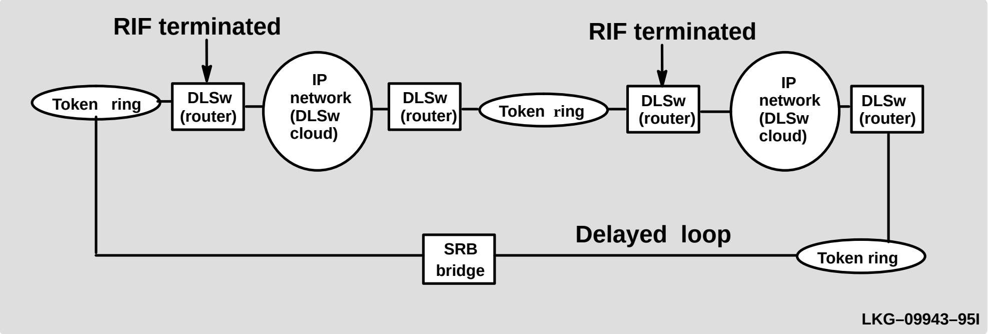
After bridging configuration is completed according to the ASRT configuration procedures described in the *Bridging Configuration Guide*, DLSw must be configured according to the procedures in the “Configuring DLSw” chapter in this document. When using the **enable dlsw** command, a source routing segment (LAN IDentification) number is assigned to DLSw. This segment number must be as follows:

- Unique in the SRB domain.
- All the DLSw routers connected to the DLSw must be configured with the same segment number.

DLSw terminates the data link for the SNA connections including the Routing Information Field (RIF). This allows you to double the hop count limitation for the source routing domain. When designing network topologies with DLSw, it is recommended that you do not allow multiple DLSw clouds. However, if necessary, you must be careful to not allow delayed loop.

Figure 14–12 illustrates an example of SRB and DLSw network topology with delayed loop.

Figure 14–12 SRB Bridge with DLSw over Token Ring

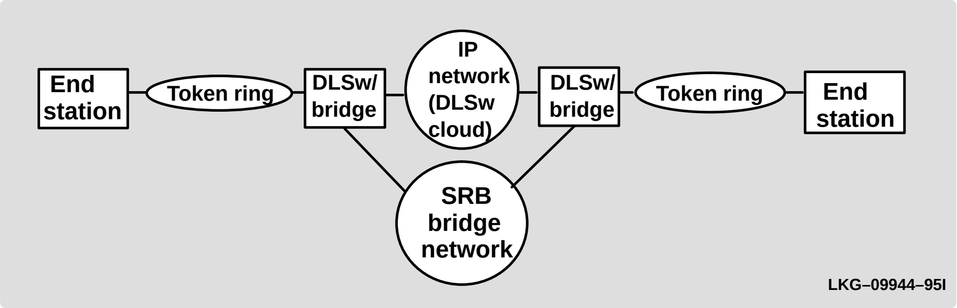


DLSw with Parallel Bridging

In a network topology built entirely of source routing bridges, you can configure DLSw in parallel to a SRB bridge in two ways.

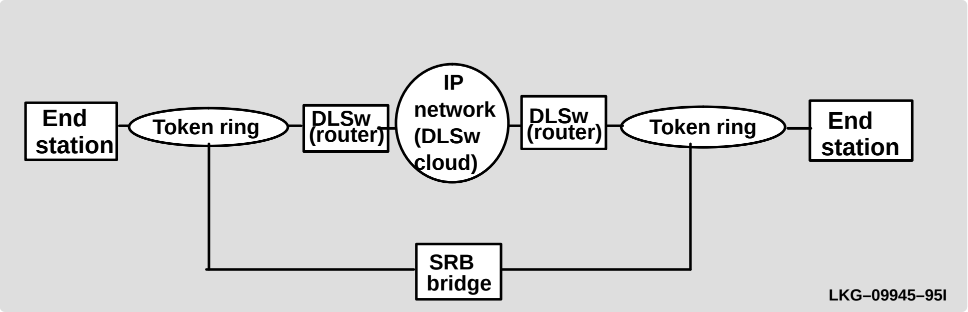
In Figure 14–13, DLSw is primarily used to connect SNA end stations only. You can customize traffic flow by configuring SAP filters in the bridge configuration. This causes the bridge portion of the router to discard the SNA frames, thus allowing only DLSw to process them. In the absence of the SAP filters, the end station can select from different paths (that is, Bridge or DLSw). It is more likely that the bridge path is always selected over the DLSw path, because the initial response is faster on the bridge path. This defeats the purpose of DLSw, which provides the link termination for time sensitive data link connections of SNA.

Figure 14–13 Parallel Bridging with DLSw and a Bridge Network



In Figure 14–14, there is a separate bridge path consisting of external SRB bridge entities, in parallel to DLSw path. In order to eliminate competing bridge paths in a particular topology, use SAP filtering of SNA frames in the bridges.

Figure 14–14 Parallel Bridging with DLSw and SRB Bridge



Note: In transparent bridge environment, parallel bridge path must not be allowed because it causes endless looping of the bridged frames. Look for and eliminate the possible parallel bridge paths.

DLSw Interoperability with IBM 6611

In order for Digital's DLSw to interoperate with the IBM 6611 DLSw implementation there are various configuration issues that need to be fully considered. The following sections provide an overview of some of the configuration issues to be addressed and also summarize some features of the current Digital DLSw implementation that are not fully interoperable with DLSw on the IBM 6611. In all cases, these issues are described based on interoperability testing performed with the IBM 6611's MPNP V1.2 Software Version, and may not be appropriate for older or more recent MPNP Software versions.

Bridge Configuration

The following are bridge configuration issues:

- The LAN identification (Segment number) of the DLSw must match on both the Digital and IBM 6611 routers. If a mismatch exists, enter the Digital Configurator (Task 6) and select the DLSw protocol. The **set srb** command can then be used to set a Segment Number value that matches the IBM 6611 equivalent.

- The maximum MTU value that can be used for the Bridge Frame is 2100 bytes. This is the largest value currently supported by the IBM 6611. If MTU values less than 2100 are specified, it is important that the configured values match on both the Digital and IBM 6611 routers.

- Currently we interoperate with the IBM 6611 only for SNA traffic over DLSw. The Digital router does not yet support NetBIOS traffic over DLSw. There is, however, a proprietary Digital solution that permits NetBIOS traffic to bridge through an IP tunnel.

IP Configuration Issues

The following are IP configuration issues:

- The is and peer/peer DLSw group feature that enables Digital DLSw neighbors to dynamically find each other is not interoperable with the IBM 6611 DLSw implementation. As a result, the DLSw's **add tcp neighbor** configuration command must be used to define the static IP addresses of adjacent IBM 6611 DLSw peers.

- The preceding interoperability restriction on the Digital DLSw group feature has implications for the selection of RIP/OSPF:

  - To utilize DLSw groups on a Digital router, the configuration of OSPF/MOSPF is also required. But since these DLSw groups are not interoperable with the 6611, it is possible to configure the Digital DLSw router with only RIP enabled and no OSPF configuration.

  - Although, OSPF and RIP can both be enabled on the Digital side, MOSPF (if selected through the OSPF configuration) is not currently supported by the IBM 6611.

  - For the IBM 6611 MPNP V1R2.0 software, the APPN network node implementation on the 6611 only appears to work with RIP.

- Within the Digital IP configuration make sure that the fill patterns configured for broadcast addresses on a given interface match their equivalent definition on the IBM 6611.

- Digital's Bandwidth Reservation System (BRS) that can be utilized to guarantee bandwidth for the transport of SNA traffic over DLSw, is not interoperable with the IBM 6611 DLSw implementation.

Although the prioritization assigned by the Digital hardware for BRS can be implemented in an outbound direction, the prioritization order is not guaranteed if intermediate IP routers do not support BRS. Also, since the 6611 does not support BRS in its end of the line, BRS is only applicable in a single direction.

TCP Related Interoperability Issues

The following are TCP interoperability issues:

1. **TCP Connection Break Detection Differences.** The Digital DLSw implementation detects that a TCP connection is broken either when a Keepalive response is not received (assuming that the Keepalive option was enabled for the connection), or when data cannot be delivered.

2. **TCP Connection Reestablishment Differences.** Once a TCP connection is broken, the Digital DLSw implementation re-establishes the TCP connection when a new DLSw SSP_CANUREACH is generated upon receipt of a DLC TEST message from an end station. The IBM 6611 may not exhibit the same behavior.

3. **Keepalive Disable/Enable Related Differences.** As indicated previously the Digital DLSw implementation permits the enabling/disabling of a Keepalive option when a TCP neighbor IP address is added (configured). Although TCP in the IBM 6611 DLSw implementation responds to Keepalive messages received on a TCP session, there is no mechanism to configure the resident 6611 TCP so as to enable the generation of TCP Keepalive messages.

4. **Maximum Number of TCP Connections Supported.** In the Digital DLSw implementation, there is no hard-coded restriction on the maximum number of TCP connections supported. As a result, the maximum number of TCP connections supported is directly related to a Digital DLSw Router’s available memory. In the IBM 6611 case, there is a hard coded internal restriction of 100 TCP connections that can be supported in the DLSw implementation.

SDLC Related Interoperability Issue

The Digital DLSw implementation currently functions as SDLC Primary only. The IBM 6611 MPNP V1R2.0 software also provides support for SDLC Primary only.

DLSw Related Interoperability Issues

The following are DLSw related interoperability issues:

- The Digital DLSw implementation does not support generation of SSP_IAMOKAY message (SSP Message Type ’x1D’) while IBM 6611 DLSw implementation is supported. This SSP message is undocumented in RFC 1434, and is silently discarded by the Digital DLSw implementation upon receipt.

- The IBM 6611 DLSw implementation processes SSP_ENTER_BUSY/EXIT_BUSY messages received from the Digital DLSw implementation but does not generate similar flow control related SSP messages.

- The Digital DLSw implementation does support the user-defined SSP_TEST_CIRCUIT_REQ message (SSP message type ’x7A’) that is generated by an IBM 6611 DLSw router functioning as an APPN network node. Upon receipt of this message, the Digital DLSw implementation returns the user-defined SSP_TEST_CIRCUIT_RSP message (SSP message type ’x7B’). This response is expected by the IBM 6611 DLSw router’s APPN network node implementation.

Miscellaneous Interoperability Issues

The following are miscellaneous interoperability issues:

- The IBM 6611 DLSw implementation is not in full compliance with the Control Message and Information Message formats defined in RFC 1434. For example, the IBM 6611 chooses to fill bytes in reserved fields with ’xFF’ values, whereas the Digital DLSw implementation zeros these fields whenever SSP Control or Information messages are transmitted. Keep these differences in mind whenever a Wide Area Sniffer is being used to monitor DLSw SSP messages flowing across a DLSw WAN connection.

- If a problem is encountered when trying to establish a DLSw connection initiated by the IBM 6611, check the IBM 6611 configuration to ensure that MAC address filtering has not been inadvertently enabled for an associated source or destination MAC address.

- Although RFC 1434 does not specifically address the issue of orphan DLSw sessions (for example, DLSw sessions that remain in a DLSw circuit established state with no subsequent activity), both the Digital and IBM 6611 DLSw implementations resolve this issue by providing orphan DLSw session timeouts. DLSw sessions that remain inactive while in DLSw circuit established state for longer than 30 seconds are eliminated by both implementations.

Basic LLC Connection Establishment

This section provides an overview of a typical LLC connection establishment, and the initial route discovery phase of the LLC connection. This section describes the following:

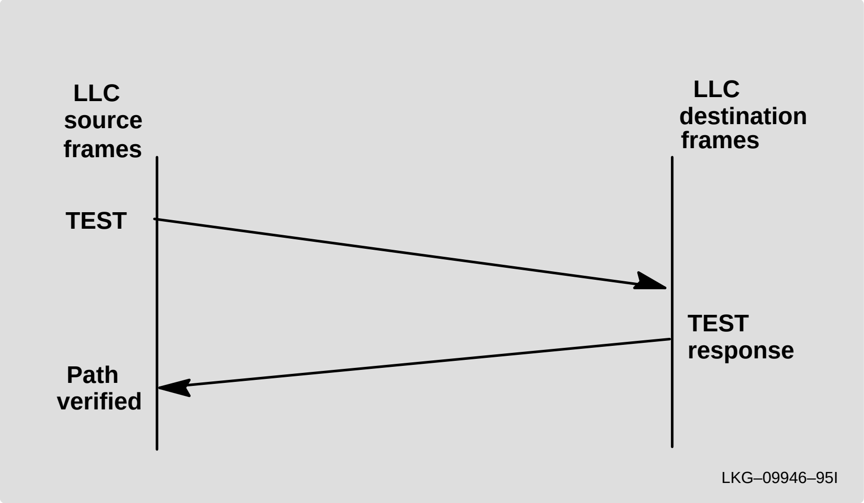
- LLC message types
- LLC data exchange
- LLC error handling
- LLC connection termination

When an SNA end station needs to send a packet to a specified destination, it must first establish an LLC connection. The type of LLC connection used is LLC Type 2.

An LLC sender discovers the route to an LLC peer by transmitting a TEST command frame to the destination. The destination returns a TEST response frame to ensure the sender that the LLC-to-LLC path is established.

Figure 14–15 shows LLC connection verification. If there are intermediate bridges in the path, a Routing Information Field (RIF) forms in the TEST command/response frames. This RIF is then used for sending out subsequent packets.

Figure 14–15 LLC Connection Verification



LLC2 Message Types

Table 14–1 shows different message types LLC2 uses when communicating with another LLC2 peer.

Table 14–1 LLC2 Message Types and Descriptions

Message	Description
UNNUMBERED: **(U-FORMAT)**	Unnumbered or non-sequenced. These frames are used to create, maintain, and terminate a session.
SABME	Set Asynchronous Balance Mode Extended (SABME). Establishes logical connection.
DISC	Disconnect (DISC). Terminates a logical connection.
UA	Unnumbered Acknowledgement (UA). Acknowledges a SABME or DISC was received.
DM	Disconnect Mode (DM). Verifies that the logical connection is terminated.
FRMR	Frame Reject (FRMR). Indicates that the LLC destination has received a bad frame and that this condition continues to exist until the connection is reset by a SABME or a DISC terminates the connection.
SUPERVISORY: **(S-FORMAT)**	Is used for acknowledgement and flow control. This frame indicates one of three commands, RR (receive ready), RNR (receive not ready), and REJ (reject).
RR	Receive Ready (RR). acknowledges all frames before the specified sequence number. It is also used to request the LLC sender to begin transmission again after an RNR.
RNR	Receive Not Ready (RNR). Acknowledges all frames before the specified sequence number and asks the LLC sender to suspend transmission. Transmission begins again only after an RR is received.
REJ	Reject (REJ). Indicates that the transmission of the I-frames starts again at the specified sequence number.
(I-FRAME)	Information (I) Format. Contains the information the sender is transmitting.

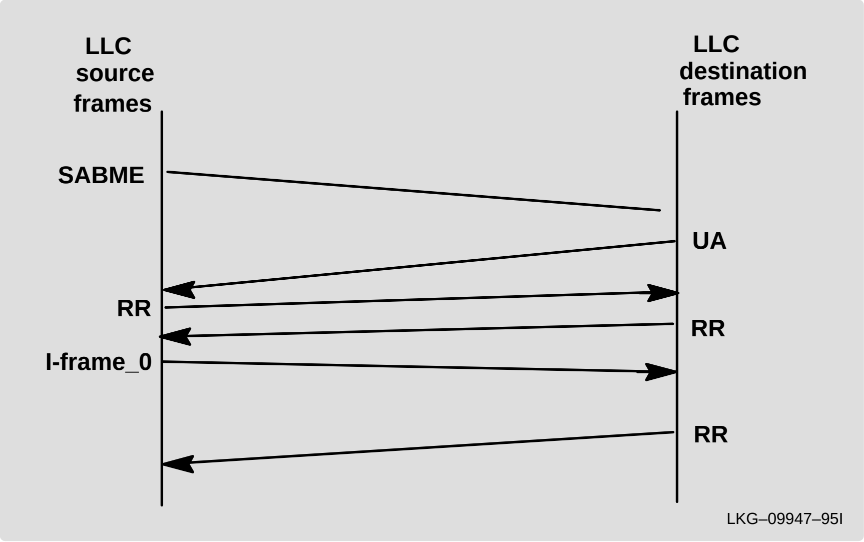
LLC2 Connection Establishment and Data Exchange

The LLC attempting to initiate a connection transmits a SABME requesting a Type 2 connection. The UA frame is sent by the destination accepting the Type 2 connection request (SABME). Normally, LLC2 connection establishment is pursued only after a path between source and destination end station was discovered through initial TEST/XID frame exchange.

LLC2 is a connection-oriented data link that is established between the source and destination before the data exchange takes place.

Figure 14–16 illustrates how an LLC connection is established.

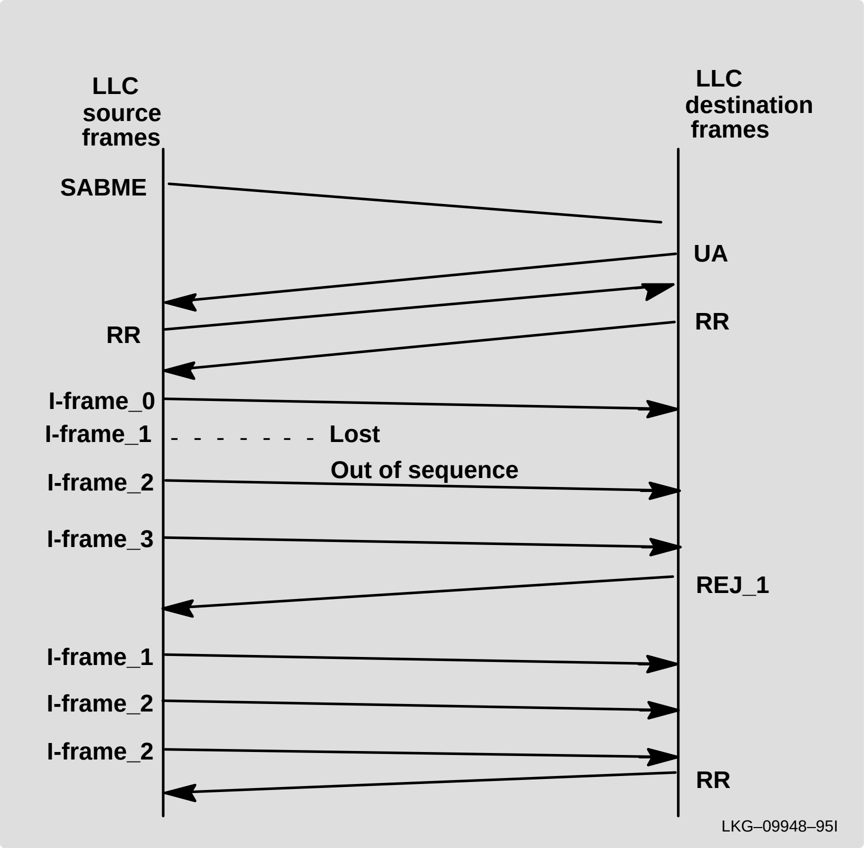
Figure 14–16 LLC Connection Establishment



LLC2 Error Handling and Connection Termination

If an LLC peer detects an error while processing a frame, either a REJ or FRMR frame is sent to the sending LLC peer. If a REJ is sent, the sender must retransmit the frame or frames beginning with the frame that is being rejected (see Figure 14–17).

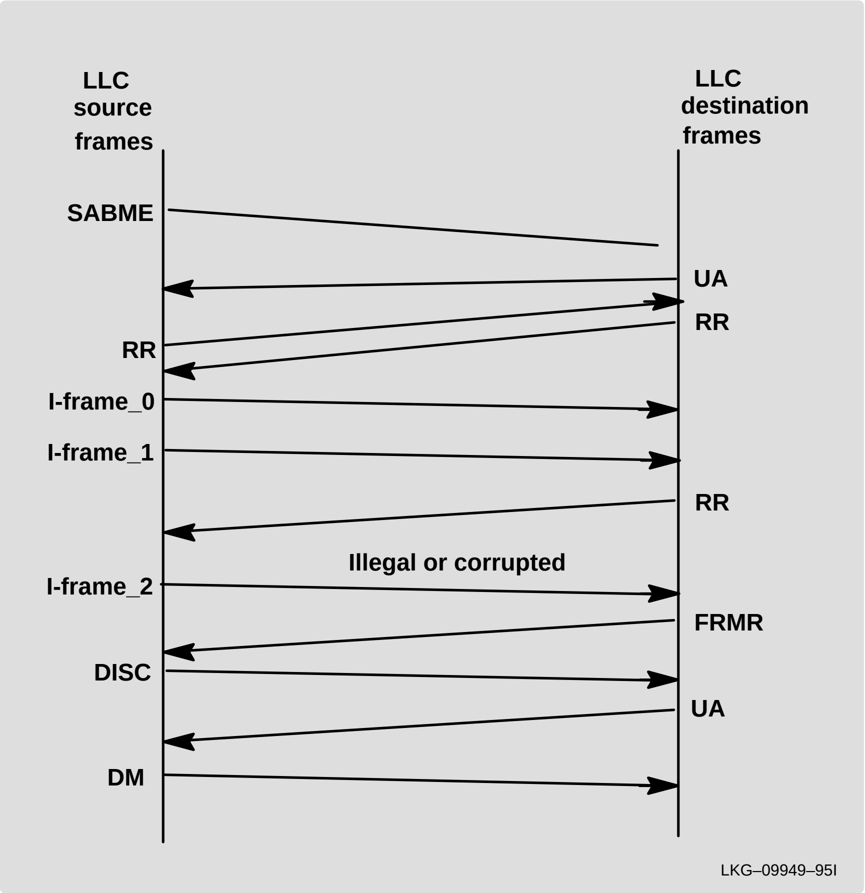
Figure 14–17 LLC Error Notification



If a FRMR is sent, the destination is notifying the sender that the frame is corrupted or protocol is violated and the link must be either reset with a SABME or terminated with a DISC (see Figure 14–18).

Note: DISC is also used to normally terminate a connection.

Figure 14–18 LLC FRMR Notification and Termination



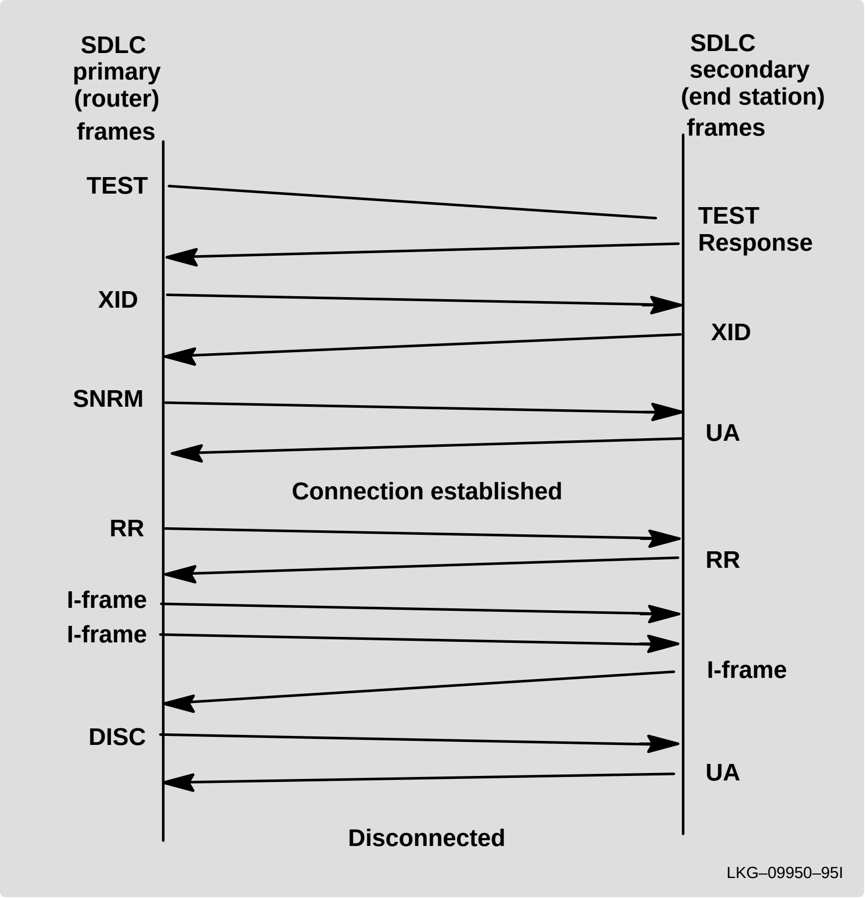
Note: Connection Termination or Connection Reset can occur normally without a preceding error condition.

Basic SDLC Connection Establishment

This section provides you with an example of a typical SDLC connection establishment. Establishing an SDLC connection is similar to the sequence for LLC2, except that all flow control is initiated by the primary end station.

Figure 14–19 illustrates how an SDLC connection is established, data is exchanged, and a connection terminated. This figure further shows I-Frames continuing implicit acknowledgments.

Figure 14–19 SDLC Connection Verification



15

The V.25 *bis* Network Interface

This chapter describes the V.25 *bis* network interface.

V.25 *bis* Overview

The V.25 *bis* network interface software allows you to interconnect routers through the general (switched circuit) telephone network using V.25 *bis* modems. The V.25 *bis* interface initiates and accepts switched-circuit connections, either on demand, automatically from restart, or on command by the operator. This allows you to reduce communications costs because you use only resources required for specific needs such as infrequent or off-peak data transfers or for WAN-Restoral operation. It also offers the flexibility of connecting to several destinations.

Each V.25 *bis* network interface consists of one serial line interface that is connected to a V.25 *bis* modem and a set of dial circuits. The V.25 *bis* interface operates as the DTE (Data Terminal Equipment) and the modem operates as the DCE (Data Communications Equipment).

Note: You can also use the V.25 *bis* interface to connect a router to an Integrated Services Digital Network (ISDN) using a V.25 *bis*-compliant ISDN terminal adapter.

V.25 *bis* Serial Line Interface

The V.25 *bis* serial line interface establishes connections with a peer router using the V.25 *bis* modem. The serial line interface accepts or initiates connections on command from the dial circuits. Once the connection is established, the serial line interface transparently passes data to and from the dial circuit.

Routing protocols cannot communicate directly with a V.25 *bis* serial line interface. These protocols refer to the dial circuit and not to the V.25 *bis* interface.

Dial Circuits

Dial circuits are a set of potential point-to-point network connections that are mapped to V.25 *bis* serial line interfaces. There are three types of dial circuits:

- Static circuits (or leased lines).
- Switched circuits that dial on demand and hang up after a specified idle time.
- WAN-Restoral circuits that are used only when a leased line fails.

Add a dial circuit for each potential destination. You can map multiple dial circuits to one serial line interface. Each dial circuit appears as a normal serial line network, running either Serial Line Protocol (SLP) or standard Point-to-Point (PPP) protocol. The protocols are configured to operate over the dial circuits.

Addressing

To place a telephone call, you need to specify the telephone number of the destination. To identify yourself to the destination, you need to specify your own telephone number. For V.25 *bis*, telephone numbers are called network dial addresses and, for convenience, they are given names called network address names that you can use to identify the telephone number.

When you set up a V.25 *bis* serial line interface, you add addresses for each potential destination as well as for your own telephone number, which is called the local network address. When you configure a dial circuit, you set up the dial circuit with one of the destination addresses.

Circuit Contention

If one dial circuit is using the V.25 *bis* circuit, other dial circuits are not able to use it. The V.25 *bis* device (modem, terminal adapter, and so forth) can have only one outstanding call at a time. Packets sent by protocols on dial circuits that cannot connect to the V.25 *bis* interface are dropped.

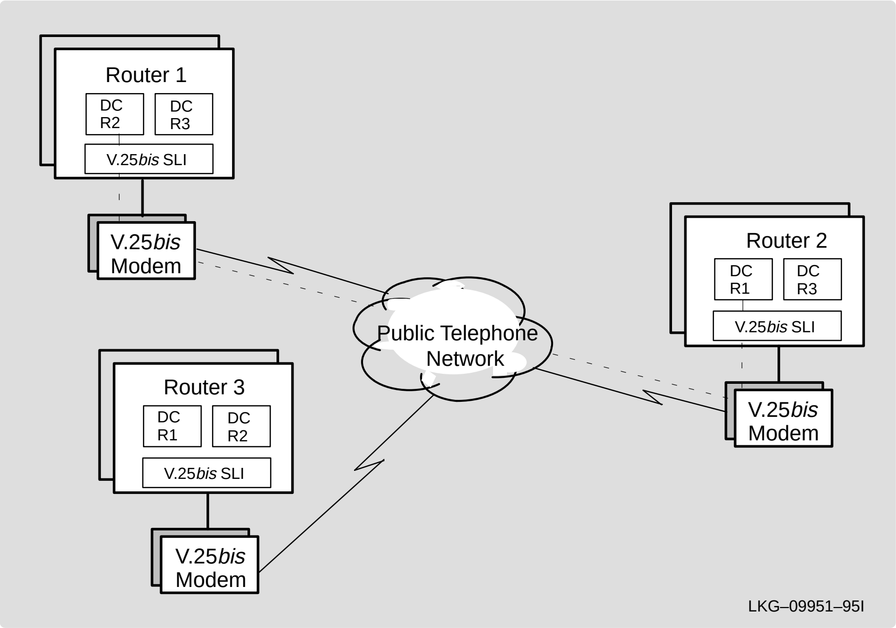
Call Verification

Digital’s V.25 *bis* implementation uses a proprietary caller-ID protocol to match incoming calls to specific dial circuits. The caller-ID protocol uses the inbound and outbound destination addresses in the dial circuit configuration to match the dial circuit that places the call to the dial circuit that receives the call. The caller-ID protocol is a brief identification protocol initiated by the caller and answered by the dial circuit that receives the call. If the caller does not provide the caller-ID message, the call may be rejected.

Sample V.25 *bis* Configuration

In Figure 15–1 there are three routers that have dial circuits (DCs) configured as potential connections between each router through the V.25 *bis* serial line interfaces (SLI) and the V.25 *bis* modems. In this example, dial circuit DCR2 in Router 1 has established a connection to dial circuit DCR1 in Router 2. To establish the connection, DCR2 initiated a call through the V.25 *bis* serial line interface to its destination address (DCR1).

Figure 15–1 Example of a V.25 *bis* Connection



Requirements and Restrictions

This sections outlines requirements and restrictions for the V.25 *bis* interfaces.

V.25 *bis* Modem Requirements

You must have a V.25 *bis* modem that supports the following circuits as outlined in the ITU/CCITT V.25 *bis* 1988 specification:

- Circuit 106–Ready to Send (Clear to Send)
- Circuit 107–Data Set Ready
- Circuit 108/2–Data Terminal Ready
- Circuit 125–Calling (Ring) Indicator

Note: The V.25 *bis* interface does not work with DCE (null modem) cables.

V.25 *bis* Interface Restrictions

The following are some V.25 *bis* interface restrictions:

- You cannot boot the router over a V.25 *bis* link.
- Bandwidth reservation is not supported on a V.25 *bis* link.
- Because X.21 supports only two circuits, the V.25 *bis* interface does not support X.21.
- Digital's implementation of V.25 *bis* supports addressed call mode (also known as circuit 108/2 mode) rather than direct call mode. Addressed call mode provides access to multiple remote destinations through one DCE.

PSL and PPP Configuration Requirements

The V.25 *bis* interface supports the PSL and PPP protocols. You need to consider the following when you configure PSL or PPP with V.25 *bis*:

- The V.25 *bis* interface predefines clocking as external and encoding as NRZ. The DCE controls the clock speed. The V.25 *bis* interface ignores those parameters in the PPP or PSL configuration.

- The V.25 *bis* interface does not enforce transmit delay counters that you set in the PSL or PPP configurations.

- Make sure that the PSL frame size of the dial circuits on all routers is set to at least 602. (The default is 2048.) The PSL protocol requires an initial exchange of messages of this size.

- Do not enable pseudo-serial-ethernet on the dial circuit.

Cost Control Over Demand Circuits

Dial circuits always appear to be in the Up state to the routing protocols. Most protocols send out periodic routing information that could cause the router to dial out each time the routing information is sent over dial-on-demand circuits. Configure IP and OSI to use only static routes, and disable the routing protocols (RIP, OSPF) over the dial circuits. This inhibits periodic routing updates. If you are using IPX, configure it for low-frequency updates and change updates so that it sends broadcast messages only at specified times or when there is a problem.

16

The WAN-restoral Interface

This chapter provides reference information about a Wide Area Network (WAN) restoral interface.

WAN-restoral Overview

The WAN-restoral process provides a backup for a serial link in the event of a failure whether the failure is on the interface or the leased line.   This restoral process is entirely transparent to the forward protocols (for example, Point-to-Point Protocol, and Serial Link) except for transparent data lags and a possible change in data rate due to the slower speed secondary. All routing information, Protocol connections, and so on are maintained.

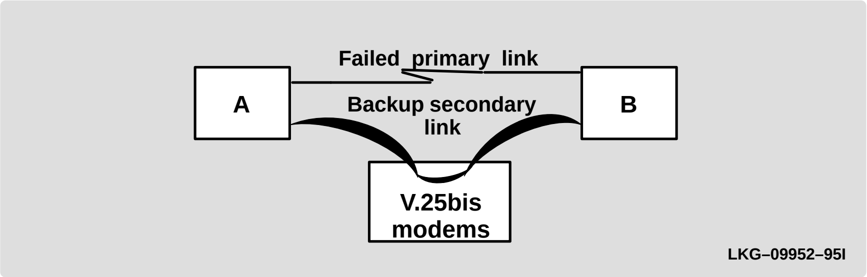
The backed-up link (that is, the one that is normally in use) is referred to as the *primary.* The link that replaces it is referred to as the *secondary.* The process involves:

- Detecting the primary link failure
- Switching to the secondary alternate link
- Detecting the primary link recovery
- Switching back to the primary link

WAN-restoral Using Two Routers

Figure 16–1 shows how a failed WAN link to another router is backed up. **Router A** normally uses a leased line to communicate with **router B**. If there is a failure in the primary WAN link connection between two routers, the secondary reconnects through a dial-up link using V.25 *bis* modems (for V.25 *bis* information see Chapter 15). When the primary WAN link connection is recovered, the secondary link disconnects automatically (this process may take a few seconds).

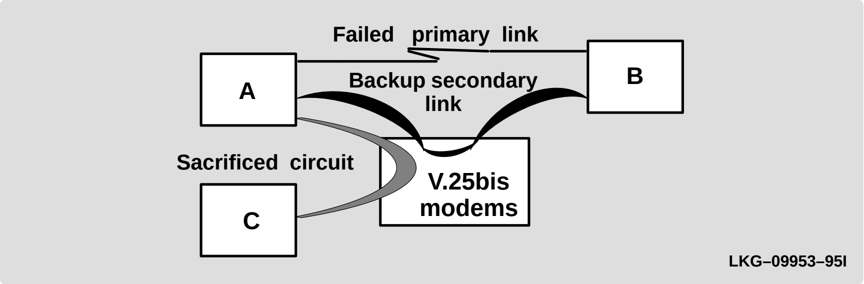
Figure 16–1 WAN Link Restoral with Two Routers



WAN-restoral Switched Connection to a Third Router

Figure 16–2 shows how backup is performed when there is a switched connection to a third router in the network. **Router A** normally uses a leased line to communicate with **router B** and a switched link to communicate with **router C.** If there is a failure in the primary WAN link between **router A** and **router B**, you must manually ***disable*** the dial circuit to **router C** to allow the WAN-restoral mechanism to use the switched circuit modem (V.25 *bis* interface) to communicate to **router B**. When the primary WAN connection is recovered, the secondary link disconnects. To reestablish the sacrificed link connection, you must use the console **test** command to restart the dial circuit on **router A** and **C**.

Figure 16–2 WAN Link Restoral with Three Routers



WAN-restoral Configuration

The following steps provide you with an overview of tasks needed to configure WAN-restoral.

1. Assign a primary to a secondary interface
2. Enable the restoration of the primary through the specific-secondary (that is, for all secondaries that are specifically enabled)
3. Enable WAN-restoral

WAN-restoral Commands

WAN-restoral provides you with configuring and monitoring commands.

The WAN-restoral configuration commands allow you to:

- Identify a backup dial circuit for a particular primary serial link. You can only assign one secondary interface to a primary.
- Remove the assignment of a secondary (backup) interface to the primary interface.
- Disable the WAN-restoral interface functionality or disable the restoral of a particular primary interface by its assigned secondary interface.
- Enable the WAN-restoral interface functionality or enable the restoral of a primary link by a secondary link.
- Display configuration information on one or all restored circuits using the list command.

The WAN-restoral monitoring commands allow you to:

- Clear accounting and statistical information.
- Disable the WAN-restoral interface functionality or to disable the restoral of a particular primary interface by its associated secondary interface.

- Enable the WAN-restoral interface functionality or enable the restoral of a primary link by a secondary link.

- List monitoring information on one or all restored circuits.

Configuration and monitoring commands are described in the *Network Interface Operations Guide*.

Secondary Dial Circuit

The secondary link that provides backup to the primary, must be a dial circuit configured to support the same link layer protocol, using the same link layer parameters.

At the current time, only one type of dial circuit is supported. It is mapped to a serial line interface that connects to the telephone network through V.25 *bis* modems. In the future, additional physical layer switched circuit interfaces are supported.

The link layers supported by WAN-restoral are currently limited to Serial Link Protocol and the Point-to-Point Protocol (PPP). To determine which link layer protocol is destined for the primary, use the **list devices** command at the `Config>` prompt. It shows the data link as part of the device configuration.

For information on how to set the secondary dial circuit refer to the *Network Interface Operations Guide.*

Point-To-Point Protocol

If the primary link is running PPP, the secondary dial circuit link must also be configured for PPP. Use the **set data-link** command at the `Config>` prompt to define the link layer protocol, and use the **encapsulator** command at the `Dial Circuit>` prompt in the dial circuit configuration to access its link layer PPP configuration.

Keep all of the “upper layer” PPP configuration exactly the same between the primary link and the secondary dial circuit.

Serial Link Protocol

If the primary link is configured as Serial Link Protocol, the secondary assigned as its backup must also be so configured using the **set data-link** command at the `Config>` prompt. Use the **encapsulator** command at the `Dial Circuit>` prompt in the dial circuit configuration to access its link layer. The only link layer parameter of significance for a Serial Link dial circuit is the frame-size, which must be the same as the primary.

For information on Serial Link protocol refer to the *Network Interface Operations Guide*.

17

MAC Filtering

This chapter explains the MAC Filtering feature available for the bridging router.

MAC Filtering Overview

MAC Filtering is a feature that lets you specify packet filters to be applied to packets during processing. Filters are a set of rules applied to a packet to determine how the packet is handled during bridging.

Note: MAC Filtering currently affects only bridged traffic.

During the filtering process, packets are either processed, filtered, or tagged during bridging. The following explains these actions:

- **Processed** – Packets are permitted to pass through the bridge unaffected.
- **Filtered** – Packets are not permitted to pass through the bridge.
- **Tagged** – Packets are allowed to pass through the bridge but are marked with a number in the range of 1 to 64 based on a configurable parameter.

A MAC Filter is made up of three objects:

- **Filter-item** – A single rule that is to be applied to a MAC address field of a packet. The result of applying the rule is either a TRUE (the match was successful) or FALSE (the match was not successful) condition.
- **Filter-list** – Contains a list of one or more filter-items.

- **Filter** – Contains a set of filter-lists.

MAC Filtering Parameters

You may specify some or all of the following parameters in creating a filter:

- Source MAC address or destination MAC address.
- Mask to be applied to the packet's source or destination MAC address.
- Interface number.
- Input/Output designation.
- Include/Exclude/tag designation.
- Tag value (if the tag designation is given).

Filter-Item Parameters

The following parameters are used to construct a filter-item:

- Address: <*Hex-Address*>
- Address Type: SOURCE or DESTINATION
- Tag: <*Tag-value*>
- Address Mask: <*Hex-Mask*>

Each filter-item specifies an address that is compared with one of the MAC addresses in the packed. The address type specifies which address in the packet, SOURCE or DESTINATION, to use in the comparison.

The address mask is a MAC address entered in hex that is used in comparing the packet's addresses. The mask is applied to the SOURCE or DESTINATION MAC address of the packet before comparing it against the filter-item MAC address.

The mask must be of length equal to the MAC address and specifies the bytes to be used in a logical “and” operation with the bytes in the packet’s MAC address before the equality comparison to the filter-item MAC address is made. If no mask is specified, it is assumed to be all 1’s.

Filter-List Parameters

The following parameters are used to construct a filter-list:

- Name: <*ASCII-string*>
- Filter-Item List: *filter-item1, ..., filter-item*
- Action: INCLUDE, EXCLUDE, TAG(n)

A filter-list is built from one or more filter-items. Each filter-list is given a unique name.

Applying a filter-list to a packet consists of comparing each filter-item in the order by which the filter-items appear in the list. If any of the filter-items in the list return a TRUE condition then the filter-list returns its designated action.

Filter Parameters

The following parameters are used to construct a filter:

- filter-list Names: <*ASCII-string*>, ..., <*ASCII-string*>
- Interface Number: <*IFC-number*>
- Port Direction: INPUT or OUTPUT
- Default Action: INCLUDE, EXCLUDE or TAG
- Default Tag: <*Tag-value*>

A filter is constructed by associating a group of filter-list names with an interface number and assigning an INPUT or OUTPUT designation. The application of a filter to a packet means that each of the associated filter-lists is applied in the order in which they appear in the filter to packets being received (INPUT) or sent (OUTPUT) on the specified numbered interface.

When a filter evaluates a packet to an INCLUDE condition, the packet is forwarded. When a filter evaluates a packet to an EXCLUDE condition, the packet is dropped. When a filter evaluates to a TAG condition, the packet being considered is forwarded with a tag.

An additional parameter of each filter is the default action that is the result of a non-match for all of its filter-lists. It can be set to either INCLUDE, EXCLUDE or TAG. In addition, if the default action is TAG, a tag value is also given.

A Word About Tags

MAC Address filtering is handled by a joint effort between bandwidth reservation and the MAC Filtering feature (MCF) using *tags.* A user with bandwidth reservation is able to categorize bridge traffic, for example, by assigning a tag to it.

The tagging process is done by creating a filter item in the MAC Filtering configuration console and then assigning a tag to it. This tag is then used to set up a bandwidth class for all packets associated with this tag. Tag values must currently be in the range of 1–64.

Up to 5 tagged MAC addresses can be set from 1 to 5. TAG1 is searched for first, then TAG2, and so forth, to TAG5.

Once a tagged filter is created in the MAC Filtering configuration process, it is then assigned a class and priority in the Bandwidth Reservation configuration process. The **tag** command is then used in the Bandwidth Reservation process to reference the tag.

Tags can also refer to “groups” as in the example of IP Tunnel. Tunnel endpoints can belong to any number of groups, and then packets are assigned to a particular group through the tagging feature of MAC address filtering.

For more information about tagging, see Chapter 13.

18

NETBIOS Name Caching

This chapter describes the NETBIOS name caching feature available on the bridging router.

Overview

The NETBIOS name caching feature enables the bridging router to significantly reduce the number of Name-Query frames that leave an originating ring and are forwarded through a bridge.

NETBIOS uses a 16 character name for host identification. In the first step of a data transfer, a client resolves a server’s name to a physical address and route. Resolving starts by the client sending a spanning tree explorer group frame called a Name-Query. The remote host’s response is either a directed (datagram) or all routes (reliable session initiation) individually addressed frame called a Name-Query-Response.

Unfortunately, NETBIOS client stations normally do not save the results of a name resolution. As a result, each time a datagram is sent or connection initiated, a Name-Query frame is generated and sent to all NETBIOS stations on all bridged segments. Adding to network overhead, the client also transmits six Name-Query requests in half second intervals before waiting for a response.

There are also two other explorer group frames (Add-Name and Add-Group-Name) that are also sent in groups of six. These frames are not directly related to name caching.

NETBIOS name caching feature helps to reduce the number of Name-Query (and associated) frames by the following methods:

- Name caching (saving NETBIOS names)
- Filtering duplicate frames

Duplicate-frame filtering works on Add-Name, Add-Group-Name and Name-Query frames. The filtering of Name-Query requests does not interfere with name caching; it just limits the number of requests that name caching receives.

Name Caching

When a client attempts to resolve a server’s name, it sends out a Name-Query frame. The server sends back a corresponding Name-Query-Response frame containing the MAC address of the desired server, as well as the route to that server.

With name caching, the bridging router acts on behalf of the client host by maintaining a database of names and routes. Each time the bridge receives a Name-Query-Response frame, the MAC and route are extracted from it and entered into the database.

When the bridge receives a Name-Query, it checks to see if the name being queried is already in its database. If it is, the frame is converted from a STE (spanning tree explorer) frame to an SRF (specifically routed) frame. A timer on the entry invalidates the database information if the server does not respond within a configurable amount of time.

Another timer exists for each overall entry to entirely remove an entry that a client or server has not referenced within a configurable amount of time.

Name Cache Processing

The following steps occur during the name cache Name-Query processing:

1. The database is searched using the server’s name as a key.
2. If the name is not in the database, an entry is created and an STE frame is sent. Frame processing ends.
3. If the entry indicates that a response is received since the last request, the interval time is updated and the frame is converted to SRF using the entry’s information.

4. If a Name-Response from the server has not been received in the required time, then the entry's information is invalidated and the frame is sent as is (that is, STE frame).

The following steps occur for Name-Query-Response processing:

1. If the name is in the database, the update timestamp and flag are updated, indicating a response is received.
2. If the name is not in the database, nothing happens.

Duplicate-Frame Filtering

This section explains duplicate frame filtering. As mentioned, three frame types are typically sent in groups of six:

- Name-Query
- Add-Name
- Add-Group-Name

Duplicate frame filtering uses a timer to allow only one instance of each type of frame to be forwarded through the bridge in the amount of time set by the user.

This process uses a separate database from the one used in Name Caching. Duplicate frame database entries contain the client's MAC address and three time stamps – one for each of the mentioned frame types. Duplicate-frame filtering is processed before name caching.

19

The DVMRP Protocol

This chapter describes the DVMRP (Distance Vector Multicast Routing Protocol).

Introduction

DVMRP is a routing protocol being run on UNIX workstation comprising the MBONE. The resulting UNIX routing daemon is called “mrouted.”

DVMRP does the following for the bridging router:

- Allows the bridging router to be used in the MBONE in addition to (instead of) UNIX workstations

- Allows MOSPF domains to be substituted for collections of DVMRP tunnels, easing bandwidth demands

DVMRP Modes

DVMRP/MOSPF can be run in one of the following three modes. The modes are listed below in order of increasing functionality.

- **Mode 1** – The bridging router functions as a regular DVMRP router. It acts like a UNIX workstation running the mrouted program. The bridging router can run DVMRP on its LAN interfaces and support tunnels (encapsulated only).

- **Mode 2** – The bridging router can join a MOSPF domain to the mbone through one or more (encapsulated) DVMRP tunnels.

In this mode, selected internal MOSPF networks are advertised into the mbone's DVMRP system. A subset of the DVMRP sources are advertised into the MOSPF system as OSPF AS external LSAs.

A MOSPF domain joined to the MBONE in this way receives the benefit of MOSPF's pruning. Therefore, only those multicast datagrams with active group members are forwarded into the MOSPF domain.

- **Mode 3** – The bridging router runs as an MBONE router using an MOSPF domain as a transit "network." In this mode, you run DVMRP over MOSPF, as if the entire MOSPF domain were a single LAN. This mode lets you replace a collection of DVMRP tunnels with an MOSPF domain. This actions results in a decrease in multicast traffic.

DVMRP/MOSPF Interaction

The bridging router can be used to join a MOSPF domain to the MBONE (see **Mode 1** above), or can allow a MOSPF domain to glue together pieces of the MBONE (see **Mode 2** above). This is done by allowing a limited exchange of information between DVMRP and MOSPF.

For the MBONE to forward multicast datagrams with sources that belong to the MOSPF domain, the router must advertise certain internal MOSPF networks through DVMRP.

To avoid increasing the DVMRP routing table size by advertising all internal networks, only those networks specified as area address ranges in the OSPF **add range** configuration commands are advertised. In particular, this allows aggregation of sources before advertising to DVMRP.

Note: It is possible to advertise two routes into the MBONE, one route being a subset of the other. Avoid this because mrouted does not handle such routes and non-deterministically discards one of the routes.

Conversely, for the MOSPF domain to forward multicast datagrams that are sourced from elsewhere in the MBONE, the DVMRP routing information, which consists of a collection of reachable sources, must be leaked automatically into the MOSPF domain in the form of OSPF AS external LSAs. As long as both DVMRP and MOSPF are enabled in the bridging router, this happens automatically and does not need to be configured.

The way that the precise set of DVMRP routes is leaked into OSPF is the following:

- The router looks at its entire collection of DVMRP sources. If more than half of the sources are reachable through non-MOSPF interfaces or DVMRP tunnels, a multicast default is imported (that is, an AS external LSA having destination 0.0.0.0 and the MC-bit set in its options field).
- If not, each DVMRP source is imported in a separate AS external LSA.

  If the router does not otherwise advertise an AS external LSA for the DVMRP source (that is, it is not the best router to use for unicast traffic destined for the source), it specifies a cost of LSInfinity in the AS external LSA.

Note: Increase the *maximum number of external advertisements* that you configure in OSPF by the number of DVMRP sources (currently around 500). Do this in all routers, not just the ones running DVMRP.

Running DVMRP over MOSPF

When using a MOSPF domain to join DVMRP tunnels, DVMRP is actually run over MOSPF. In this case, a DVMRP interface (VIF) named ”MOSPF” is automatically created and DVMRP probes and updates are sent to the multicast address 239.0.0.1 (a group that all routers simultaneously running DVMRP and MOSPF join).

The bridging router forwards 239.0.0.1 throughout the MOSPF domain, but never forwards 239.0.0.1 over DVMRP interfaces or tunnels. The address 239.0.0.1 was not registered with the Network Interface Card (NIC).

Tunnels to Internal MOSPF/OSPF Destinations

It is possible to configure DVMRP tunnels to internal OSPF destinations. When this is done, the software assumes that MOSPF connectivity does not exist to the destination. Therefore, the DVMRP tunnel is preferred over the possible MOSPF path. This can be enforced on reception of packets, but not on packet transmission.

Therefore, when MOSPF connectivity does exist to the other end of the DVMRP tunnel, the tunnel endpoint receives multiple copies of all multicast datagrams.

Supported and Unsupported DVMRP Features

By running as a replacement for a UNIX workstation running mrouted, DVMR supports the following subset of mrouted functionality:

- DVMRP can run natively on all LAN interfaces. Tunnels are also supported, although only the encapsulated version. Source routed tunnels are NOT supported.

- MBONE mapping queries are supported. Therefore, the DVMR responds to the UNIX **mrinfo** program. It returns a version of 1.

- DVMRP pruning is NOT supported. However, MOSPF does support pruning. Connecting a MOSPF domain to the mbone through one or more DVMRP tunnels produces the desired result.

- The rate limiting supplied by mrouted is NOT supported. However, bandwidth reservation feature (available on serial line) has a class for IP multicast traffic.

  Creation of this traffic class lets you limit the amount of forwarded multicast traffic to a fixed percentage of available serial line bandwidth. Conversely, multicast traffic can also be guaranteed a dedicated percentage of the line’s resources.

  IP access controls also apply to multicast traffic. For example, you can set up the Digital router so that it only carries certain vat sessions, and no others.

20

Bridging Features

This chapter describes bridging features that are available with the Adaptive Source Routing Transparent (ASRT) Bridge.

Bridging Tunnel

The bridge tunnel (encapsulation) is another feature of the ASRT bridge software. By encapsulating packets in industry-standard TCP/IP packets, the bridging router can dynamically route these packets through large IP internetworks to the destination end stations.

End stations see the IP path (the tunnel) as a single hop, regardless of the network complexity. This helps overcome the usual 7-hop distance limit encountered in source routing configurations. It also lets you connect source routing end stations across non-source routing media, such as Ethernet networks.

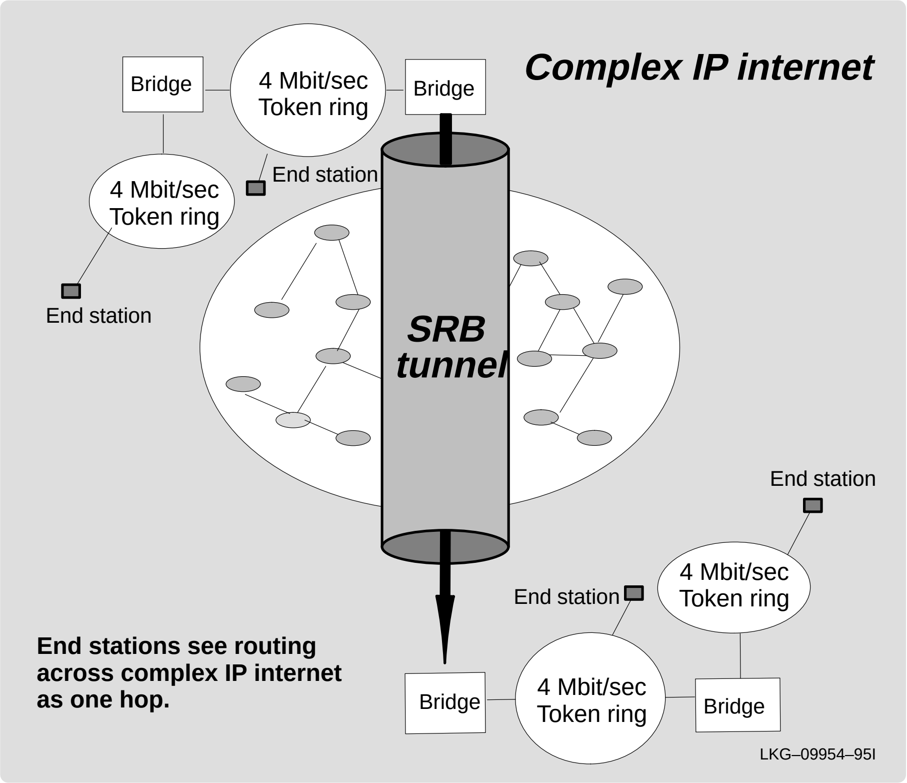
The bridging tunnel also overcomes several limitations of source routing including the following:

- Large amounts of overhead that source routing causes in wide area networks (WANs).
- Source Routing’s sensitivity to WAN faults and failures (if a path fails, all systems must restart their transmissions).

With the bridge tunnel feature enabled, the software encapsulates packets in TCP/IP packets. To the router, the packet looks like a TCP/IP packet. Once a frame is encapsulated in an IP envelope, the IP forwarder is responsible for selecting the appropriate network interface based on the destination IP address. This packet can be routed dynamically through large internetworks without degradation or network size restrictions. End stations see this path or tunnel, as a single hop, regardless of the complexity of the internetwork.

Figure 20–1 shows an example of an IP internetwork using the tunnel feature in its configuration.

Figure 20–1 Example of the Bridge Tunnel Feature



End stations see routing across complex IP internet as one hop.

The tunnel is transparent to the end stations. The bridging routers participating in tunneling treat the IP internet as one of the bridge segments. When the packet reaches the destination interface, the TCP/IP headers are automatically removed and the inner packet proceeds as a standard source routing packet.

Encapsulation and OSPF

A major benefit of the encapsulation feature is the addition of dynamic routing. Dynamic routing (using OSPF, Integrated IS-IS, or RIP) offers the following benefits when used with encapsulation:

- **Least-Cost Routing** – The shortest path is chosen for the tunnel.

- **Adaptive Routing** – The tunnel path automatically adapts to route around failed links.

With dynamic routing, if a line or brouter fails along the path, then the tunnel bridge or router automatically reroutes traffic along a new path. If a path is restored, the tunnel automatically updates to the best path. This rerouting is completely transparent to the end stations. For more information about dynamic routing for IP, refer to Chapter 2.

TCP/IP Host Services (Bridge-Only Management)

The Bridging Router also supports TCP/IP Host services that let you configure and monitor a bridge when IP routing functions are disabled. This option gives you the following capabilities:

- Management through SNMP.
- Telnet server functionality.
- Downloading and uploading of configurations through the TFTP protocol.
- TFTP neighbor boot functionality.
- IP diagnostic tools of ping and traceroute.
- Control of the device through SNMP sets and the telnet client.

When viewed from the bridge's console interface, TCP/IP Host Services is handled as a new protocol having its own configuration and monitoring consoles. These prompts are accessed through the **protocol** command in the `Config>` and `+` (GWCON) consoles.

Bridge-only management functionality is activated by assigning an IP address to the bridge and enabling TCP/IP Host Services. This IP address is associated with the bridge as a whole, instead of being associated with a single interface. When booting over the network, the bridge's IP address and a default gateway can be learned automatically. Default gateway assignments may also be user-configured.

Bridge-MIB Support

For Bridge Management through SNMP, the brouter supports the Bridge-MIB as specified by RFC 1286. The entire Bridge-MIB is implemented except for the following:

- The forwarding database table for transparent bridges (dot1dTpFdbTable).
- The static (Destination-Address Filtering) database table (dot1dStaticTable).
- The newRoot and topologyChange traps.

The entire Bridge-MIB is read-only.

NETBIOS Filtering

ASRT Bridging performance can be enhanced by an included feature called NETBIOS filtering. NETBIOS filtering lets you configure specific filters through the router configuration process. These filters are sets of rules applied to NETBIOS packets to determine if the packets are bridged (forwarded) or filtered (dropped). These filters can be applied to the following aspects of the NETBIOS packets:

- Host name fields in the packets
- Arbitrary fields (bytes) in the packets

NETBIOS Filtering using host names lets you select packets with specific NETBIOS host names to be bridged or filtered. Another NETBIOS filtering mechanism, byte filtering, specifies certain NETBIOS packets to be bridged or filtered based on arbitrary fields (bytes) in the NETBIOS packets.

Filtering is useful because NETBIOS traffic can contain a high proportion of broadcast packets. Unfiltered NETBIOS traffic can take up a large percentage of network bandwidth (particularly on low-speed WAN interfaces) and greatly reduce the overall performance of WAN interfaces and the network. Configuring NETBIOS filters helps to correct this problem.

A NETBIOS filter (host name or byte) is made up of three parts:

- The actual filter
- Filter lists
- Filter items

Each filter is made up of one or more filter lists. Each filter list is made up of one or more filter items. Each filter item in the filter list of a filter is evaluated against a packet in the order in which the filter items were specified. When a match between a filter item and a packet is found, the filter list containing the filter item evaluates to the configured value of the matching filter item (Inclusive or Exclusive). This evaluation determines whether the packet matching the filter item is bridged or filtered.

NETBIOS Filtering Using Host Names

NETBIOS Filtering using host names lets you select packets with specific NETBIOS host names to be bridged or filtered. When you specify that packets with a particular NETBIOS host name (or set of NETBIOS host names) are to be bridged or filtered, the source name or destination name field of the following NETBIOS packet types is examined:

- ADD_GROUP_NAME_QUERY (source NETBIOS name field is examined).
- ADD_NAME_QUERY (source NETBIOS name field is examined).
- DATAGRAM (destination NETBIOS name field is examined).
- NAME_QUERY (destination NETBIOS name field is examined).

Host name filter lists specify NETBIOS names to be compared with source or destination name fields in the four different types of NETBIOS packets just described. The result of applying a host name filter list to a NETBIOS packet that is not one of those four types is Inclusive.

When configuring NETBIOS Filtering using host names, you specify which ports the filter is applied to and whether it is applied to input or output packets on those ports. Only NETBIOS Unnumbered Information (UI) packets are considered for filtering. Filtering is applied to NETBIOS packets that arrive at the router for either Source Route Bridging (all RIF types) or Transparent Bridging.

When specifying a NETBIOS host name in a filter, you can indicate the 16th (last) character of the name, as a separate argument, in its hexadecimal form. If this is done, the first 15 bytes of the name are taken as specified and the 16th byte (if any is specified) is determined by the final argument. If fewer than 16 characters are specified (and no 16th byte is specified), then the name is padded with ASCII blank characters up to the 15th character, and the 16th character is treated as a wildcard.

When a specific NETBIOS host name is evaluated, that name is compared with only certain fields of certain NETBIOS packets. NETBIOS host names in filter items might include a “?” wildcard character at any point in the NETBIOS host name, or a “*” as the final character of a NETBIOS host name. The “?” matches any single character of a host name. The “*” matches any 1 or more characters at the end of a host name.

NETBIOS Filtering Using Bytes

Another filtering mechanism, byte filtering, is also available to let you specify which NETBIOS packets to be bridged or filtered. With byte filtering, you specify certain NETBIOS packets to be bridged or filtered based on arbitrary fields in the NETBIOS packets. In this case, all NETBIOS packets are examined to determine if they match the configured filtering criteria.

The following are filter items that you can be specify to be evaluated in a byte filter:

- An offset from the beginning of the NETBIOS header.
- A byte pattern to match on.
- An optional mask to apply to the selected fields of the NETBIOS header.

The mask, if present, must be of equal length as the byte pattern, and specifies bytes that are to be logically ANDed with the bytes in the NETBIOS header before the header bytes are compared to the hex pattern for equality. If no mask is specified, it is assumed to be all 1’s. The maximum length for the hex pattern (and hence the mask) is 16 bytes (32 hexadecimal digits).

When configuring NETBIOS Filtering using specific bytes, you also specify which ports the filter is applied to and whether it is applied to input or output packets on those ports.

Building a Filter

Each filter is made up of one or more filter lists. Each filter list is made up of one or more filter items. Each filter item in the filter list of a filter is evaluated against a packet in the order in which the filter items were specified. When a match between a filter item and a packet is found, the filter list containing the filter item is evaluated for a configured indicator (Inclusive or Exclusive) to determine whether the packet matching the filter item is bridged or filtered.

If no filter items in the filter list produce a match, the filter list is evaluated for its default indicator value (Inclusive or Exclusive). If the filter contains multiple filter lists then each filter list of the filter is evaluated. Once all filter lists in a filter are evaluated, then the filter as a whole is given an Inclusive or Exclusive indication. The packet is then bridged or filtered based on that indication.

A filter item is a single rule applied to a particular field of a NETBIOS packet. The result of the application of the rule is either an Inclusive (bridge) or an Exclusive (filter) indication. The following lists the filter items that can be configured with NETBIOS Filtering (the first two items are host name filters, the last two items are byte filters):

- Include <NETBIOS host name> <optional 16th character (hex)>
- Exclude <NETBIOS host name> <optional 16th character (hex)>
- Include <decimal byte offset into NETBIOS hdr>
  <hex pattern starting at that offset><hex mask>
- Exclude <decimal byte offset into NETBIOS hdr>
  <hex pattern starting at that offset><hex mask>

Part of the specification of a filter list is to indicate whether packets that do not match any of the filter items in the filter list to be bridged (Included) or filtered (Excluded). This is the default action for the filter list. The default action for a filter list is initially set to Include, but this setting can be changed by the user.

Simple and Complex Filters

A simple filter is constructed by combining one filter list with a router port number and an input/output designation. This indicates that the filter list is to be applied to all NETBIOS packets being input or output on the given port. If the filter list evaluates to Inclusive, then the packet being considered is bridged. Otherwise, the packet is filtered.

A complex filter can be constructed by specifying a port number, an input/output designation, and multiple filter lists separated by one of the logical operators AND or OR. The filter lists in a complex filter are evaluated strictly left to right, and each filter list in the complex filter is evaluated. Each Inclusive filter list result is treated as a TRUE and each Exclusive filter list result is treated as a FALSE. The result of applying all the filter lists and their operators to a packet is a TRUE or FALSE, indicating that the packet is bridged or filtered. Each combination of input/port or output/port can have at most one filter associated with it.

Pseudo Serial Ethernet

Pseudo Serial Ethernet is an optional mode of operation that provides for the encapsulation of any routed protocol on a bridging router proprietary serial line, to be forwarded within an Ethernet encapsulated frame over the same serial line. This allows the protocol to communicate with a pure bridge on the opposite end of the serial line.

When enabled, this mode makes the serial lines appear as an Ethernet interface to the configured routing protocols. The handler uses Ethernet (or IEEE 802.3, as appropriate) encapsulations, thus limiting the protocols to the maximum Ethernet frame size. These Ethernet frames are then sent and received as bridged Ethernet frames on the serial line. Any frames arriving on the routed protocol code points from the serial lines are ignored, and bridged Ethernet frames are passed to the bridging or routing forwarders as appropriate.

This encapsulation is normally not necessary with the bridging routers at both ends of the serial line, since both can be configured to route the same set of protocols over the same serial line.

Multiple Spanning Tree Protocol Options

The ASRT Bridge lets you extend Spanning Tree protocol options to cover as many configuration options as possible. The next sections provide information about these features.

Background: Problems with Multiple Spanning Tree Protocols

Bridging technology employs different versions of spanning tree algorithms to support different bridging methods. The common purpose of each algorithm is to produce a loop-free topology.

In the spanning tree algorithm used by Transparent Bridges (TB), Hello BPDUs and Topology Change Notification (TCN) BPDUs are sent to well known group addresses on all participating media (token ring, Ethernet, FDDI). Tables are built from this exchanged information and a loop free topology is calculated.

Source Routing Bridges (SRB) transmit Spanning Tree Explorer (STE) frames through SRB bridges to determine a loop free topology. The algorithm sends Hello BPDUs to well known functional addresses. Since TCN BPDUs are not used by SRB bridges, the port state setting created as a result of this spanning tree algorithm does not affect All Route Explorer (ARE) Frame and Specifically Routed Frame (SRF) traffic.

In bridging configurations using IBM 8209 Bridges, a different spanning tree method is used to detect parallel 8209 bridges. This algorithm uses Hello BPDUs sent as STE frames to IEEE 802.1d group addresses on the token ring. On the Ethernet, Hello BPDUs sent as transparent frames to the same group address are used. This method allows 8209s to build spanning trees with Transparent Bridges and other IBM 8209 bridges. It does not participate in the SRB spanning tree protocol, however, and Hello BPDUs sent by SRBs are filtered. There is no way to prevent the 8209 from becoming the root bridge. If the 8209 bridge is selected as the root then traffic between two Transparent Bridge domains may have to pass through token ring/SRB domains.

STP/8029

The STP/8029 bridging feature is available to allow you to further extend the Spanning Tree protocol. Previously, SRB bridges allowed only manual configuration of a loop-free tree over the token ring . This was the only mechanism to prevent loops in the case of parallel SR-TB bridges. With the addition of the STP/8029 feature, the following spanning tree algorithm combinations are possible:

- **Pure Transparent Bridge (TB)** – IEEE 802.1d Spanning Tree protocol is used.

- **Pure Source Routing Bridge (SRB)** – SRB Spanning Tree protocol is used.

- **Transparent and Source Routing Bridges as separate entities** – IEEE 802.1d Spanning Tree protocol is used for TB and SRB Spanning Tree protocol is used for SRB.

- **ST-TB Bridge** – IEEE 802.1d Spanning Tree protocol is used for TB ports and IBM 8209 BPDUs on SRB ports are used to form a single tree of TBs and SR-TBs. SRB Hello BPDUs are allowed to pass on the SR domain but are not processed.

  IBM 8029 bridges filter such frames but this is allowed as it is a 2-port bridge with the other port being a TB port.

- **Pure SRT Bridge** – *Only* IEEE 802.1d Spanning Tree protocol is used. SRB Hello BPDUs and IBM 8209 BPDUs are allowed to pass but are not processed.

- **ASRT Bridge** – IEEE 802.1d Spanning Tree protocol is used to make a tree with TBs and SRT bridges. “8209-like” BPDUs are also generated on all SR interfaces.

  These BPDUs are processed as soon as they are received. This causes two BPDUs to be generated and received on all SR interfaces. Since both BPDUs carry the same information, there is no conflict of port information. This lets the ASRT bridge create a spanning tree with IBM 8209 and SR-TB bridges along with other TBs and SRT bridges.

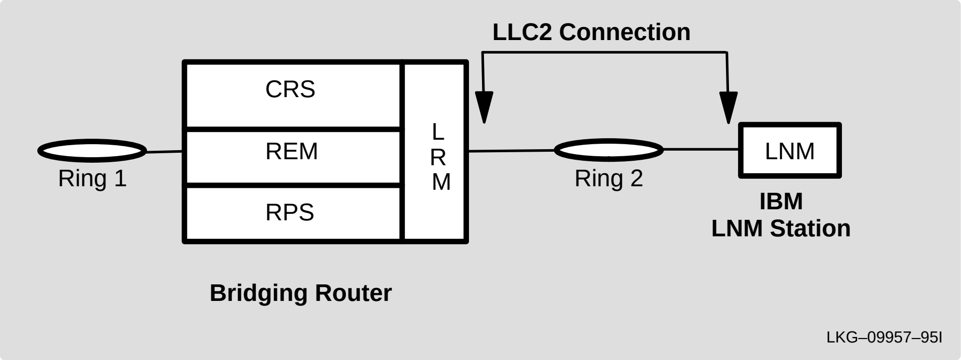
LAN Network Manager Functionality

LAN Network Manager (LNM) functionality lets you manage token-ring networks interconnected by source route bridges. LNM lets you monitor the operation of rings, bridges and individual ring stations.

Information collected by software agents on the bridging router are made available to LNM management stations. More specifically, LNM agents forward collected information through another agent called the LAN Reporting Mechanism (LRM), a proprietary IBM protocol. Information forwarding is done through an LLC2 connection (explained below) to a LAN Network Manager station.

Figure 20–2 illustrates the connection between the bridging router, LNM agents, and the LNM station.

Figure 20–2 LNM Station and LNM Agents



Specific LNM Agents and Functions

The LNM agents and their functions include the following:

- **Configuration Report Server (CRS)** – Reports ring topology changes and ring station status to LNM.
- **Ring Error Monitor (REM)** – Collects error reports from ring stations and analyzes them. When thresholds are exceeded, REM may forward error information to LNM.
- **Ring Parameter Server (RPS)** – Services requests from ring stations for ring parameter information including ring number, the soft error report timer value, and the physical location.
- **LAN Reporting Mechanism (LRM)** – Controls the establishment of reporting links from LNM stations to the bridge agents. Also manages the transfer of information to and from the other agents over these links.

The following sections describe each LNM agent in more detail.

CRS

At the request of LNM, the CRS agent obtains and forwards ring station status to LNM. It also can be used to set ring station parameters and remove a station from the ring.

Configuration information generated by ring stations is forwarded to LNM. When LNM requests the status of a ring station, CRS builds and sends MAC frames to the station to obtain the information. CRS then sends the following to the ring station:

- Request Ring Station Address MAC frame.
- Request Ring Station State MAC frame.
- Request Ring Station Attachments MAC frame.

When the ring station replies, the information is put into a properly formatted LLC2 frame and forwarded to LNM.

A ring station may be removed from the ring by CRS at the request of LNM. To remove a station, a Remove Station MAC frame is sent to the station to be removed. CRS will return a response to LNM indicating the success or failure of the removal.

When CRS receives a Report New Active Monitor MAC frame it forwards the information to LNM. When a Report NAUN Change MAC frame is received, this information is also reported. The CRS agent has its own functional address that ring station MAC layers may use to forward MAC frames to CRS.

RPS

The RPS agent concerns itself with the insertion of ring stations onto the ring. When a ring station is newly inserted into the ring the following occurs:

- The new station sends a Request Initialization MAC frame to the Ring Parameter Server (RPS) for that ring. This MAC frame includes some information about the station.
- RPS responds to this MAC frame with an Initialize Ring Station MAC frame that contains the ring number and the interval of time to wait between sending Report Soft Error MAC frames. The information gleaned from the Request Initialization frame is passed to LNM so that it may maintain a database of all ring stations on the ring.

- RPS also responds to a request for status that is sent from LNM. The ring number, RPS version information and the soft error report timer value are returned to LNM.

The RPS function has an associated functional address that is used for receiving the MAC frames that other ring stations send to it.

REM

The REM agent observes the operation of the attached token ring by looking for hard errors and soft errors. It then reports these to LRM and aids in isolating the cause of the errors. The following occurs during hard error detection:

- Hard errors are detected on the ring by the receipt of Beacon MAC frames.

- Stations in the fault domain attempt to correct the problem by possibly removing themselves from the ring.

- REM determines if the hard error condition was corrected or not and then reports to LNM the result of the hard error condition.

The following occurs as REM monitors soft errors:

- Soft Error MAC frames are sent periodically by ring stations to REM to inform it of the counts of the number of times various intermittent faults occurred. Some examples of these types of errors are CRC errors and frequency errors.

- When the number of soft errors for a station exceeds some threshold value, the indication that this has occurred is sent by REM to LNM.

- REM also monitors the Report Soft Error MAC frames for receiver congestion conditions. Receiver congestion indicates that a ring station discarded frames due to a shortage of receive buffers.

- If the number of times a station reports receiver congestion exceeds a certain threshold, REM reports this condition to LNM. When the receiver congestion condition returns to normal levels, LNM is notified that the receiver congestion condition has ended.

LRM

LRM controls the connection of LNM to the agents. LRM establishes reporting links between itself and each LNM that is connected. A reporting link is an LLC2 connection between LNM and LRM.

All communication between LNM and the agents is done through a reporting link. LRM passes management data to and from the appropriate agents to the reporting links. Up to four reporting links are supported. One is designated the controlling link and the other three are designated as observing links.

An LNM connected through the controlling link may perform all available operations. LNMs connected by observing links may only perform a limited subset of the available operations.

LNM Configuration Restrictions

The LNM agent and the LNM station always assume that messages are being passed on a two-port model. LNM is enabled, however, on a per bridge port basis to be consistent with the existing Digital configuration.

This is what is meant by a multi-port configuration. LNM can be enabled on any source-routing token ring bridge port. In other words, an instance of LNM is created for each port upon which LNM is enabled.

With the exception of a two token ring configuration, the other port in a two-port model is always designated by a false address. This address corresponds to something that is not a real token ring interface. This could be a virtual ring, for example, or a serial line.

Only in the case where the Digital bridge has exactly two source routing token ring ports will the other port in the two-port model bridge actually be a real token ring with a real address.

Note: To obtain the MAC addresses needed to configure the LNM Manager, you must enter: **`list lnm ports`**

LBS (LAN Bridge Server) is partially implemented. LBS can report packets forwarded and packets discarded performance data statistics when requested by the manager station. Remote configuration updates from the manager station are not supported.

Logical Link Class 2 Support

In LANs, the data link layer is comprised of two sublayers: the medium access control (MAC) and the link layer control (LLC). LLC provides two types of services:

- **LLC1 (Type 1)** – an unacknowledged connectionless service
- **LLC2 (Type2)** – a set of connection-oriented service

LLC2 provides capabilities for the following

- Initiating new data link connections.
- Managing data link connections.
- Exchanging data in sequential order (in a guaranteed fashion).
- Executing a level of flow control on the established connections.
- Terminating link connections upon request from the service user or unrecoverable link errors.

The LLC sublayer adheres to the IEEE 802.5 standard.

Threading

Threading is a process where the network protocol (IPX, DNA, IP, and AppleTalk) of the token-ring end station discovers a route over segments of a Source-Routing Bridge Network.

Threading is no different than the Source Routing Bridge operation. It is how threading is implemented by the end station that is different. The following sections describe threading for IP, DNA, IPX, and AppleTalk.

IP Threading with ARP

IP end stations use ARP REQUEST and REPLY packets to discover a RIF. Both IP end stations and the bridges participate in the route discovery and forwarding process. The following steps describe the IP threading process.

1. An IP end station maintains an ARP table and a RIF table. The MAC address in the ARP table is used as a cross-reference for the destination RIF in the RIF table. If a RIF does not exist for that specific MAC address, the end station transmits an ARP REQUEST packet with an ARE (All Routes Explore) or a STE (Spanning Tree Explore) onto the local segment.

2. All bridges on the local segment capture the ARP REQUEST packet and send it over their connected networks.

   As the ARP REQUEST packet continues its search for the destination end station, each bridge that forwards it adds its own bridge number and segment number to the RIF in the packet. As the frame continues to pass through the bridged network, the RIF compiles a list of bridge and segment number pairs describing the path to the destination.

   When the ARP REQUEST packet finally reaches its destination, it contains the exact sequence of bridge and segment numbers from source to destination.

3. When the destination end station receives the frame, it places the MAC address and its RIF into its own ARP and RIF tables. If the destination end station receives any other ARP REQUEST packets from the same source, that packet is dropped.

4. The destination end station then generates an ARP REPLY packet including the RIF and sends it back to the source end station.

5. The source end station receives the learned route path. The MAC address and its RIF are then entered into the ARP and RIF tables. The RIF is then attached to the data packet and forwarded onto the destination.

6. Aging of RIF entries is handled by the IP ARP refresh timer.

DNA Threading

DNA end stations use ARE to discover a route. Both the DNA end stations and the bridges participate in the route discovery process and forwarding. The following steps describe the DNA threading process.

1. If there is no entry in the RIF table for the MAC address, an entry is created with the state *NO_ROUTE*. When this occurs the end station sends the data packet out with an STE attached. The STE is used for discovery without attempting to flood the network with an ARE.

2. The end station then transmits an ARE in a loop-back frame for the destination MAC address.
3. All bridges on the local segment capture the STE and loop-back frame and send it over their connected networks.

   As the packets continue their search for the destination end station, each bridge that forwards it adds its own bridge number and segment number to the RIF in the STE and the ARE. As the frame continues to pass through the bridged network, the RIF compiles a list of bridge and segment number pairs describing the path to the destination.

   When the STE and loop-back frame finally reaches the destination, it contains the exact sequence of bridge and segment numbers from the source to the destination.
4. When the destination end station receives the loop-back frame it places the MAC address and the RIF of the source station into its own RIF table. If a RIF already exists for that entry, it either updates the RIF if that previous entry is a *ST_ROUTE* (refer to step 7) or it ignores the RIF. In any case the entry state is changed to *HAVE_ROUTE*.
5. The destination end station sends the loop-back reply frame (including the specific RIF) back to the source end station.
6. The source end station receives the learned specific route path. The RIF is entered into the RIF table and the entry is changed to *HAVE_ROUTE*.
7. Packets destined for a functional address are sent with an STE. DNA end stations can create a RIF entry using this STE frame. When this happens, the state of the entry is changed to *ST_ROUTE*. This type of route is less desirable and is replaced as described in step 4.

The DNA end stations contain an independent RIF timer. When this timer expires for a specific RIF entry, an ARE in a loop-back packet is sent out to that specific destination. When the loop-back frame returns, the RIF entry is updated. If the destination end station is on the same ring and the loop-back frame contains no RIF, the loop-back packet is returned with no RIF entry.

IPX Threading

IPX end stations check each packet they receive for a RIF. If the RIF does not exist in the table, they add the RIF to the table and designate that route as *HAVE_ROUTE*. If the RIF indicates that the packet came from an end station on the local ring, the route is designated as *ON_RING*.

If the end station needs to send out a packet and there is no entry in RIF table for the MAC address, the end station transmits the data as an STE.

When the RIF timer expires, the entry in the table is cleared and is not reentered until another packet arrives containing a RIF for that entry.

AppleTalk 1 and 2 Threading

AppleTalk end stations use ARP and XID packets to discover a route. Both the AppleTalk end stations and the bridges participate in the route discovery process and forwarding. The following steps describe the AppleTalk threading process.

1. If a RIF does not exist for a specific MAC address, the end station transmits an ARP REQUEST packet with an ARE (All Routes Explore) onto the local segment.

2. All bridges on the local segment capture the ARP REQUEST packet and send it over their connected networks.

   As the ARP REQUEST packet continues its search for the destination end station, each bridge that forwards it adds its own bridge number and segment number to the RIF in the packet. As the frame continues to pass through the bridged network, the RIF compiles a list of bridge and segment number pairs describing the path to the destination.

3. When the destination end station receives the frame, it places the MAC address and its RIF into its own ARP and RIF tables and the state of the entry is designated as *HAVE_ROUTE*.   If the destination end station receives any other ARP REQUEST packets from the same source, that packet is dropped.

4. The destination end station then generates an ARP REPLY packet including the RIF and sends it back to the source end station with the direction bit in the RIF reversed.

5. The source end station receives the learned route path. The MAC address and its RIF are then entered into the ARP and RIF tables and the state is designated as *HAVE_ROUTE*. If the RIF indicates that the packet came from an end station on the local ring, the route is designated as *ON_RING*.

6. If the RIF timer expires an XID is sent out with an ARE and the state is changed to *DISCOVERING*. If no XID reply is received, the entry is discarded.

Glossary

A

ACK or acknowledge

A character or sequence of characters sent by a receiver to notify a sender the last message was received correctly.

adjacency

In OSPF, adjacency is created between neighboring routers for exchanging routing information. In OSI, adjacency refers to a directly connected ES or IS that was configured or learned through the ES-IS protocol.

address

The logical location of a terminal, node, peripheral device, or byte in memory in a network. The abbreviation for address is addr.

advance command processing

An option for the single Ethernet interface card in which a chip links commands to expedite command transmission.

AFI

Authority and Format Identifier. Part of the OSI NSAP address that specifies the format of the IDI.

all route broadcast

The process of sending a frame through every non-repeating routes in a bridged LAN.

all station broadcast

The process of addressing a frame so that every station on the ring copies the frame.

application services

Those services referred to by the upper three layers of the OSI reference model.

area

In OSI, DNA, or OSPF a routing subdomain that maintains detailed routing information about its own internal composition, while also maintaining routing information that allows it to reach other routing areas.

area address

(1) In OSI, the remainder of the NSAP address that does not contain the system ID or the selector.

(2) In OSPF, a designator in IP format.

(3) In DNA, the six most significant bits of the area and node address.

ARP

Address Resolution Protocol. An Internet protocol that dynamically binds a high level internet address to a low level physical hardware address. ARP is across a single physical network and is limited to networks that support hardware broadcast.

AS

Autonomous System (network). A collection of routers and networks that fall under one administrative authority and cooperate closely to propagate network reachability (and routing) among themselves using an interior gateway protocol of their choice. This term is used frequently ion networks that run IP.

Asynchronous Transmission

Data transmission in which each information character, or sometimes each word or small block, is individually synchronized, usually with start or stop elements (for example, bits). The gap between each character or word is not a fixed length.

AUI

Attachment Unit Interface. A connector for the Ethernet interface that attaches a workstation to a transceiver.

B

BECN

Backward Explicit Congestion Notification. A Frame Relay method where ANSI Annex D management notifies the source device that it is receiving more frames than the frame relay backbone can process. This results in congestion at the router.

BGP

Border Gateway Protocol. A routing protocol whose function is exchange network reachability information with other BGP or EGP routers located in different ASs.

BGP Speaker

A router or host that speaks BGP.

bit-oriented protocol

A protocol that begins and ends with the same bit sequence (01111110) referred to as a flag. SDLC is a bit-oriented protocol.

BMAC

Basic Media Access Controller. A component of the FDDI interface card that supports all MAC layer protocol services, such as token claim and generation, frame transmission, reception, and stripping. The BMAC also manages the token timing logic.

Border Router

A router that speaks BGP and has a connection to two or more ASs.

bridge

A protocol independent device, that operates at the data link level, that interconnects 2 or more LANs.

bridge address

Used by the Spanning Tree algorithm, the least significant 6-octet part of the bridge identifier.

bridge identifier

Used in a Spanning Tree network to uniquely identify each bridge on the network.

bridge number

The number that identifies the specific to a segment or ring.

bridge priority

The most significant 2-octet part of the bridge identifier that is used to determine the bridge on the network that becomes the root bridge.

bps

Bits per second. The rate at which data is transmitted over a communications channel.

broadcast network

A network that transmits a packet of data any computer on the network can pickup and use.

C

CCITT

International Telegraph and Telephone Consultative Committee. An international organization that sets standards for interconnection of telephone equipment and network protocols such as X.25.

checksum

An integer value computed from a sequence of octets in a packet and compared for verification. A checksum detects errors that can result when a packet is in transit.

CIR

Committed Information Rate. In Frame Relay, when the router slows data traffic at a user-defined, predetermined rate due to congestion.

CMT

Connection Management Task. A protocol function for the FDDI interface card that establishes the physical connections between the stations.

coaxial cable

A cable consisting of a central conductor surrounded by an insulator and then by another grounded conductor. The outer cable acts as a shield to prevent interference from reaching the inner conductor.

collision

An event that occurs when two computers attached to a network attempt to send a packet simultaneously.

connection

The path between two protocol modules that provides reliable stream delivery service.

CSMA/CD

Carrier Sensed Multiple Access/Collision Detection. A method of transmitting information in the LAN environment where only one transmitter is on the line at any one time. If two devices transmit simultaneously, the signals collide and transmissions temporarily cease. The Ethernet IEEE-802.3 standard uses CSMA/CD technology.

CT

Configuration Timer. A configurable timer in the OSI protocols, ES-IS and IS-IS, that invokes the sending of an ESH (End System Hello) or an ISH (Intermediate System Hello).

CTS

Clear To Send. A control line between a modem and a controller that indicates that the controller may send data.

D

designated router (IS)

(1) In OSI, the IS on a LAN that performs additional duties. A designated router generates link state PDUs on behalf of the LAN, treating the LAN as a pseudonode.

(2) In DNA this is the default router. If there is more than one router attached to a LAN, the hosts on that LAN look to that router for information.

designated Bridge

The bridge that claims to be the closest to the root bridge in the accumulated path cost.

designate port

The port ID of the designated bridge attached to the LAN.

destination

A node designated as the intended receiver of data.

distance-vector algorithm

A class of routing algorithms that emphasizes the number of hops to find the shortest-path to a destination.

DLCI

Data Link Connection Identifier. Located in the Frame Relay header, this 10-bit field is the MAC address that identifies the PVC between the user and frame relay device.

DMA

Direct Memory Access. A method to transfer data between peripheral devices and internal memory without intervention by the central processing unit (CPU).

domain

(1) A set of rules for the operation of a protocol. An OSI domain is a set of addresses administered by the same authority that all ESs and ISs within the domain must follow to insure compatibility.

(2) Domain refers the Apollo Computer networking system.

domain name

A name or label that is mapped to a 32-bit IP address that identifies a host.

dotted decimal notation

The syntactic representation for the 32-bit IP address consisting of four 8-bit numbers written in decimal with periods separating them. For example, 190.82.10.2.

DRAM

Dynamic Random Access Memory. A type of storage the computer accesses at frequent intervals.

DSP

Domain Specific Part. A part of an OSI NSAP address that determines the network addressing authority identified by the IDI.

dynamic routing

Routing that adjusts automatically to network topology or traffic changes.

E

ECN

Explicit Congestion Notification. A Frame Relay protocol mechanism that signals source and destination user devices that network congestion is occurring.

EGP

Exterior Gateway Protocol. A protocol between ASs that advertises the IP addresses of the networks. It is also the name of one specific protocol as documented in RFC 904.

EIA

Electronics Industry Association. An organization of electronics manufacturers that establishes electronic interface standards.

encapsulation

A method by which a protocol appends required information for a peer protocol.

end-to-end

Services referred to by the lower four layers of the OSI reference model. A packet forwarded from one host to another host over the network uses end-to-end services.

entity name

In AppleTalk, an entity name is an 8-bit ASCII character string that has three fields: object, type, and zone. Each of these fields is a string of not more than 32 characters.

ES

End System. A host system, in the OSI protocol, that performs the functions of all of the layers of the OSI reference model.

ESH

End System Hello. An ES originated packet that passes information to an IS.

ES-IS

The protocol that the ESs and ISs use to recognize and communicate with each other.

Ethernet

A baseband LAN technology that uses the physical and data link layers of the OSI model. Software protocols, such as TCP/IP, provide network layer functions. Ethernet includes three standards: IEEE-802.3, Version 2.0, and Version 1.0.

event

A network message that indicates some irregularity in the physical and software elements of a network. An event may be informational or it may require the user to perform a specific task.

explorer frame

Use in a source-routing bridge to discover routes.

extended entry

AppleTalk Phase 2 only. These are typically EtherTalk 2.0, FDDI, and token ring-based networks that can take advantage of the extended network configuration capabilities of AppleTalk Phase 2. Extended networks communicate by unique network number *and* node number pairs. Extended networks are also assigned a range of network numbers and all network numbers are chosen from within this range.

external peer

Usually a BGP border router located in an adjacent AS.

F

FDDI

Fiber Distributed Data Interface. A fiber optic LAN that operates at high-speed (100 Mbps).

FECN

Forward Explicit Congestion Notification. A Frame Relay method in which the router notifies the destination device that it needs to send more frames than the line speed allows. This results in congestion at the router.

flood

Transmitting a packet across each interface on a bridge or router.

fragment

The process of identifying an IP or ISO CLNP datagram into smaller pieces for transmission across a network that cannot handle the original datagram size.

frame

Informal name for a data-link PDU. Control information in the frame provides addressing, sequencing, flow and error control to the respective protocol levels.

FTP

File Transfer Protocol. An IP application protocol that provides reliable file transfers from one network device to another.

G

gateway

A device that performs 7-layer conversion of information from one protocol stack to another.

GOSIP

Government Open Systems Interconnection Profile. An OSI NSAP addressing format to interconnect US government systems.

H

HDLC

High-level Data Link Control. An ISO standard bit-oriented data link protocol that specifies the encapsulation method of data on synchronous data links.

hello protocol

A protocol that OSPF, DNA, and OSI routers use to maintain reachability.

hello/I-H-U

Hello and I-Heard-You. An EGP protocol that requests and confirms neighbor reachability.

homogeneous

Connected networks that use the bridging method.

hop

The next router a packet must travel to arrive at its destination. A hop is represented by an address or a decimal character (how many).

host

A PC, workstation, or mainframe connected to a network.

HT

Holding Timer. An OSI configurable timer that informs an ES or IS how long it is to retain the information contained in the hello message.

I

IBD

Integrated Boot Device.

ICMP

Internet Control Message Protocol. A part of IP that handles error and control messages. ICMP includes an echo request/reply function to test whether a destination is reachable and responding. ICMP messages are incorporated into the data field of an IP packet.

IDI

Initial Domain Identifier. In OSI, the IDI specifies the network addressing domain from which the values of the DSP are allocated and the network addressing authority responsible for allocating values of the DSP from that domain. For example, 0005 designates the US government as established by NIST.

IDP

Initial Domain Part. A part of the OSI NSAP address that consists of the AFI and the IDI.

IGP

Interior Gateway Protocol. A generic term that applies to interior routing protocols.

IEEE-802

A standard for interconnecting LANs using the physical and data link layers of the ISO reference model.

IIH

Intermediate to Intermediate Hello. An OSI hello message distributed between ISs. This allows an IS to determine the existence of other ISs to establish adjacencies.

Integrated IS-IS

Another name for dual IS-IS where the OSI protocol is used to build both the IP and OSI routing tables.

interface

The physical connection the router uses to connect to a network or a line.

internal peer

A BGP speaker located in the same AS.

internet

A collection of packet switching networks interconnected by gateways with protocols. This enables networks to function as a single, cooperative virtual network. When written in uppercase, Internet refers to the TCP/IP protocol it uses.

Internet Protocol (IP)

The Department of Defense (DoD) Internet standard protocol that defines the Internet datagram as the unit of information passed across the Internet. IP Corresponds to the OSI reference model layer 3 and provides connectionless datagram service.

intra-area routing

In OSPF and DNA, this term is used to describe routing within an area.

IP datagram

A packet containing IP control information that is exchanged between network entities.

IS

Intermediate System (router). An OSI reference to a system that supports the routing function of the network layer service. There are two levels of IS: 1 and 2.

ISDN

Integrated Services Digital Network. A digital network combining voice and digital network services through a single medium. CCITT controls the technical and protocol standards for ISDN.

ISH

Intermediate System Hello. An IS originated packet to an ES.

L

LAN

Local Area Network. A network that spans a small geographic area.

LCP

Link Control Protocol. In PPP, the protocol that establishes, configures, tests, and terminates a link connection.

level 1 IS

An OSI and DNA term that describes an IS that routes NPDUs directly to systems in their own area. For NPDUs outside the a level 1 area, the NPDU is routed towards a level 2 IS.

level 2 IS

In OSI and DNA an IS that routes NPDUs from one area to another within the routing domain.

link establishment packets

LCP packets that establish and configure a PPP link.

link maintenance packets

LCP packets that are used to maintain and debug the PPP link.

link state database

A database in OSPF and OSI that collects reachability information about ESs and ISs and calculates routes based on the shortest path.

link state algorithm

A class of routing algorithm that broadcasts information on the cost of reaching each of its neighbors to all routers in the network to insure a more consistent view of the network and cost of routing packets.

link termination packets

LCP packets that close a PPP link.

LLC

Logical Link Control. IEEE defined sublayer of the OSI link layer. The LLC handles error control, flow control, and framing.

LSP

Link State Packet. In OSI and OSPF, the LSP contains reachability information about systems and areas that the router knows about. This packet is flooded across the network to other routers to maintain the link state database on each router.

LSU

Link State Update. The process that is responsible for building the LSP.

loopback

Directing signals back towards a source along a communications path.

M

MAC

Medium Access Control. A medium-specific access control protocol within IEEE-802. MAC provides a set of services to ensure proper operation of the token ring including detection of, and recovery from, error conditions.

managed object

A network device that is managed by a network management protocol.

MIB

Management Information Base. A database of managed objects that is accessed from a network management protocol.

modem eliminator

A device that allows the connection of two DTE devices without the need of a modem.

MPP

Multi-Protocol Processor. A large-scale integration device for the serial interface card that acts as the interface between host and local memory and is the master to the host bus and DDLC.

MPU

Math Processor Unit. A portion of the CPU which performs mathematical operations.

multicast

A technique that allows copies of a single packet to be broadcast to a specified number of hosts.

N

NET

Network Entity Title. An OSI reference that refers to the next hop. In general, the NET is network address of the network layer itself.

network layer

Layer 3 of the OSI reference model. This layer is where all routing occurs.

node

A term that refers to a device that can access a network.

nonextended network

AppleTalk Phase 1 only. Nonextended networks refer to networks running under the router's AppleTalk Phase 1 protocol (APL). These are typically nonextended AppleTalk Ethernet 1.0, Serial Line, or LocalTalk-based networks.

NPDU

Network Protocol Data Unit. A packet that contains network layer control information and is exchanged between network entities.

NSAP

Network Service Access Point. The point where the communications capability of the network layer is made available at the layer boundary to its users. An OSI network address.

O

OBS

Optical Bypass Switch. An optional switching function that controls whether to bypass a ring when the FDDI interface card is down or removed.

OSI

Open Systems Interconnection. The ISO architecture for internetworking.

OSI Reference model

The seven layer model specified by ISO, which specifies particular network functions.

OSPF

Open Shortest Path First. A link state IGP uses between routers to exchange routing information.

P

packet

A self-contained block of information containing control and user information that is transmitted across a network

packet switching

A data transfer scheme in which information is broken into individual packets, transferred across a communications link, and reassembled at the receiving end. In a packet-switching system, the route between the sender and receiver is determined by each node through which the packet travels.

PCM

Pulse Code Modulation. A communication system technique of carrying information by converting an analog signal to digital form.

PDN

Public Data Network. A network operated to provide computer communications to the public.

PDU

Protocol Data Unit. An OSI defined packet exchanged between ESs that contains protocol control information and user data.

PHY

Physical Layer Protocol. A protocol function that links one FDDI station to another. It provides the bit clocks for each station and an elasticity buffer between the receiver and transmitter. PHY also transmits 7 line states.

physical address

The address of the interface between the MAC interface and a LAN.

PING

Packet InterNet Groper. The name of an internet program that tests the reachability of destinations by sending an ICMP echo request and waiting for a reply.

PMD

Physical Media Dependent. A function of the FDDI interface card that provides the power levels and characteristics of the optical transmitter and receiver, optical signal requirements, and bit error rates.

port priority

In the Spanning Tree algorithm, the port priority is the second 1-octet part of the port ID.

PPP

Point-to-Point Protocol. A channel or link with only two terminals whose purpose is to transmit protocol datagrams at the data link layer over serial point-to-point links.

proxy ARP

A technique in which one machine answers ARP requests intended for another by supplying its own physical address.

pseudonode

In OSI/DNA V, an imaginary node is used with the link state routing algorithm to represent the transmission medium itself. All nodes are viewed as being connected to the pseudonode with a separate point-to-point logical link.

PVC

Permanent Virtual Circuit. A Frame Relay and X.25 feature in which data traveling between to end points uses a pre-established path. A PVC gives the appearance of a permanent point-to-point connection.

R

RARP

Reverse Address Resolution Protocol. The protocol a diskless workstation uses at start-up to find its Internet address.

RIF

(1)   Ring Interface. A component of the token-ring interface card that interconnects the serial data port of the TMS380C16 to the token ring interface card connector.

(2)   Routing Information Field. A field in the token ring 802.5 header that is used by a source-route bridge to determine the path a packet must use when passing through a token-ring network segment.

ring number

A unique number that identifies a ring in a bridged network.

RISC

Reduced Instruction Set Computer. A type of processor architecture that minimizes the number of instructions performed by the processor to increase processing speed.

RIP

Routing Information Protocol. A distance-vector IGP used between routers to exchange routing information.

root bridge

The bridge with the highest priority bridge ID that is selected as the roof of the Spanning Tree. This bridge is responsible for keeping the Spanning Tree in intact.

route

A path through a series of LANs and bridges.

route designator

A ring number and bridge number in the RIF used to build a route through the network.

route discovery

The process by which a route is learned to a destination end station.

router

A device with ability to route packets from one end station to another with multiple paths between them.

routing domain

In OSI, a set of ESs and ISs that share routing information, operate according to the same routing protocol, and are contained within a single administrative domain.

routing Subdomain

A set of ISs and ESs located within the same routing domain.

S

SDLC

Synchronous Data Link Control. A bit-oriented link layer protocol that is a subset of the HDLC protocol.

seed node

The router that comes up first and verifies the configuration of the other routers. If the configuration is valid, the other routers start functioning. The seed router comes up even if there are no other routers on the network.

segment number

A number that identifies each individual LAN, such as a single token ring or a serial line.

server

A node or host that provides services to a client.

single route broadcasting

The process of sending a frame through a network such that exactly only copy of the frame appears on each ring on the network.

SMDS

Switched Multimegabit Data Service. High-speed, packet-switched, WAN networking technology.

SMT

Station Management Task. A protocol function for the FDDI interface card that controls and monitors overall station activity including initialization, activation, maintenance, and error control within each station.

SNPA

Subnetwork Point of Attachment. An OSI reference to the access point to a subnetwork topology. The same as the physical address.

source routing

A bridging mechanism that routes frames through a multi-LAN network by specifying in the frame the route it travels.

spanning tree

A bridge topology that ensures there is only one data route between any two end stations.

SRAM

Static Random Access Memory. A type of random-access memory that holds its contents without constant refreshing from the CPU.

static route

A route that is manually entered to the routing table.

subnet

In IP, a distinct network within a network. In OSI, subnet is the connection from the IS to the subnetwork.

subnetwork

Network segment. In OSI, a collection of ESs and ISs under the control of a single administrative domain and using a single network access protocol. In IP, the sharing of a particular subnet address.

subnet address

An extension of the IP addressing scheme that allows a site to use a single IP address for multiple physical networks.

subnet mask

A 32-bit address mask that is used to specify a particular subnet.

synchronous transmission

A form of data transmission in which data is sent continuously against precise time base that is shared by transmitting and receiving terminals.

system ID

The portion of the OSI NSAP address that identifies a specific system within an area.

T

T1

A long-haul transmission medium capable of transmitting information at 1.544 Mbps.

TCP/IP

Transmission Control Protocol/Internet Protocol.

TFTP

Trivial File Transfer Protocol. A simplified version of FTP that provides unreliable file transfers.

thinnet

A type of coaxial cable to run the Ethernet interface.

token ring

A network topology in which the next logical node receiving the token is also the next physical location on the ring.

Transmission Control Protocol

Corresponds to layer 4 of the OSI reference model and provides reliable transmission of data.

transmit password

A character string added to all outgoing OSI packets.

transparent bridging

A type of bridging mechanism that is invisible to each end station.

TTL

Time to Live. The amount of time an IP router holds a datagram before discarding it.

X

X.25

The CCITT standard protocol for transport level network service. X.25 supports remote login.

Z

zones

An arbitrary (user-defined) subsets or conceptual groups of nodes within two or more networks.

zone list

The zone list is a set of character strings that name the network. Each node on the network chooses one of the names from the list. Several networks can use the same zone name. A broadcast to all nodes in a zone goes to all networks that advertise that zone name.

zone name

In AppleTalk, a name given by a network manager to an arbitrary subset of networks within an internet. This name is a string of not more than 32 characters.

Index

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA
call 800–DIGITAL

In Puerto Rico
call 809–754–7575 x2012

In Canada
call 800–267–6215

In New Hampshire
Alaska or Hawaii
call 603–884–6660

ELECTRONIC ORDERS (U.S. ONLY)

Dial 800–234–1998 with any VT100 or VT200 compatible terminal and a 1200 baud modem. If you need assistance, call 1–800–DIGITAL.

DIRECT MAIL ORDERS (U.S. AND PUERTO RICO*)

U. S. SOFTWARE SUPPLY BUSINESS
DIGITAL EQUIPMENT CORPORATION
10 Cotton Road
Nashua, New Hampshire 03063–1260

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
940 Belfast Road
Ottawa, Ontario, Canada K1G 4C2
Attn: A&SG Business Manager

INTERNATIONAL

DIGITAL
EQUIPMENT CORPORATION
A&SG Business Manager
c/o Digital's local subsidiary
or approved distributor

Internal orders should be placed through the Software Services Business (SSB)
Digital Equipment Corporation, Westminster, Massachusetts 01473

*Any prepaid order from Puerto Rico must be placed
with the Local Digital Subsidiary:
809–754–7575 x2012

Distributed Routing Software
Routing Protocols Reference Guide
AA–QL2CB–TE

READER’S COMMENTS

What do you think of this manual? Your comments and suggestions will help us to improve the quality and usefulness of our publications.

Please rate this manual:

	Poor				Excellent
Accuracy	1	2	3	4	5
Readability	1	2	3	4	5
Examples	1	2	3	4	5
Organization	1	2	3	4	5
Completeness	1	2	3	4	5

Did you find errors in this manual? If so, please specify the error(s) and page number(s).

General comments:

Suggestions for improvement:

Name Date

Title Department

Company Street

City State/Country Zip Code

DO NOT CUT – FOLD HERE AND TAPE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY LABEL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

digital™

Shared Engineering Services
550 King Street
Littleton, MA 01460–1289

DO NOT CUT – FOLD HERE