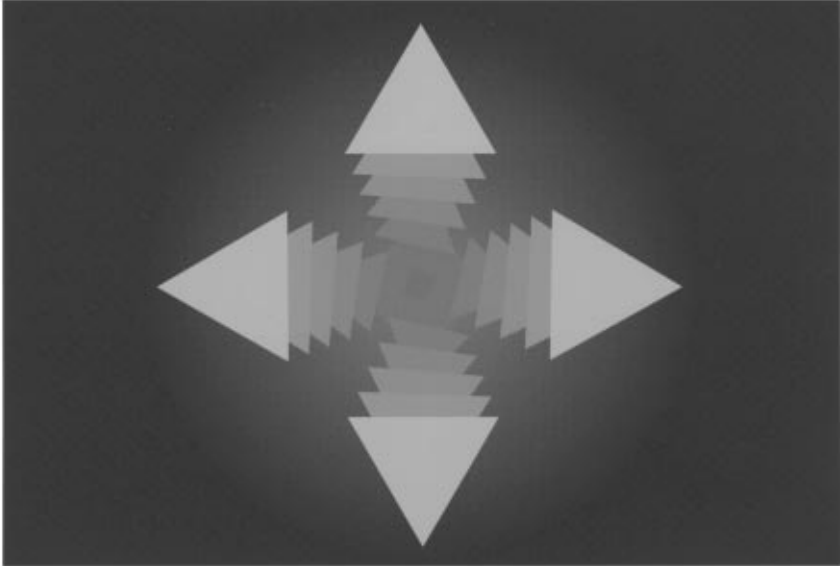


ROUTERS

OC-6993/1.0



XL Series

Reference Guide
Volume 2 of 3

Olicom XL Series Reference Guide

volume 2 of 3

Routing

Notice

The information in this document is subject to change without notice. Olicom assumes no responsibility for any damages arising from the use or inability to use this document, including, but not limited to, lost revenue, lost data, claims by third parties or other damages. For complete warranty information, refer to the product warranty card that covers this document and its product.

Copyright Notice

Olicom reserves the right to modify the information given in this publication without prior notice. The warranty terms and conditions applicable for your purchase of this equipment are given at the time of purchase. Please consult your place of purchase for details.

Publication: P/N: 710001498

Copyright© Olicom A/S, Denmark, December 1997.

All rights reserved. No part of this document may be copied, photocopied, reproduced, translated or reduced to any electronic or machine-readable format without prior written consent of Olicom.

Software Copyright Notice

The software described in this document is covered by one or more of the following copyrights:

Copyright© Olicom A/S, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997.

Copyright© Hewlett-Packard Company, 1988, 1989, 1991, 1992.

Copyright© Carnegie Mellon University, 1989.

Trademark Notice

3Com and 3+Open are trademarks of 3Com Corporation.

DEC and DECnet are trademarks of Digital Equipment Corporation.

Ethernet and XNS are trademarks of Xerox Corporation.

ClearSight is a trademark of Olicom.

ILAN is a trademark of I-LAN, Inc.

IBM is a registered trademark of International Business Machines Corporation.

Internetwork Packet Exchange (IPX) is a trademark of Novell, Inc.

Megahertz is a registered trademark of Megahertz Corporation.

NetWare and Novell are registered trademarks of Novell, Inc.

OpenView is a registered trademark of Hewlett-Packard Company.

Sprint is a registered trademark of Sprint.

Stratacom is a registered trademark of Stratacom, Inc.

VINES is a registered trademark of Banyan Systems, Inc.

Table of Contents

Routing

1. Introduction to Routing	1
2. IP Routing	5
Overview	5
Technical Discussion	6
Network Considerations	19
3. RIP	25
Overview: IGP and EGPs	25
RIP	27
RIP Filters	28
4. OSPF	35
Overview	35
Technical Discussion	37
Network Considerations	47
5. Border Gateway Protocol	55
Overview	55
Technical Discussion	56
BGP Terminology and Concepts	60
6. TCP	63
Overview	63
Technical Discussion	65
7. IPX Routing	77
Overview	77
Technical Discussion	81
Network Considerations	105
8. Protocol Independent Routing (PIR)	111
Overview	111
Technical Discussion	113
Network Considerations	117
9. Virtual Port	121
Overview	121
Technical Discussion	122
Topology Guidelines	127
SLCS considerations	130
DLSw considerations	130

10. ClearSession Protocol	131
Overview	131
Technical Discussion	132
Network Considerations	139

SNA and NetBIOS

11. Introduction to SNA and NetBIOS	141
Overview	141
SDLC/HDLC PassThrough	142
SLCS	142
DLSw	143

12. SDLC/HDLC PassThrough	145
Overview	145
Technical Discussion	146
Network Considerations	148

13. SLCS	151
Overview	152
Typical SLCS Configurations with PU 2.0 Devices	153
Timeout Avoidance through Local Termination	155
PU 2.1 and PU 1.0 Devices	158
LLC and SDLC	160
Connecting SLCS to Ports, Interfaces, and the Packet Switch Engine	164
SLCS and Bridging	165
Two Sides of a SLCS Session	166
Host, FEP, SLCS and Controller Parameters for PU 2.0 Devices	168
FID2 Segmentation, Multidrop and Group Polling	176
XID Identifiers During SLCS Session Establishment	178
NetView Support	178
Sample Complex Topology	180

14. DLSw	181
Overview	181
Typical DLSw configuration	184
Switch-to-Switch Protocol	185
DLSw partner setup	185
SNA Device Handling	186
SDLC-Attached SNA Device Handling	189
NetBIOS Devices handling	189
Benefits of Local Termination	192
Flow Control	192
DLSw Coexistence with Bridging and Routing	193
Enabling/Disabling Traffic on given XL ports	194

Virtual Port Parameters	194
DLSw Filters	195
Priority Management	202

Index

Illustrations

Figure 1. Class A Address Format	7
Figure 2. Class B Address Format	7
Figure 3. Class C Address Format	7
Figure 4. Class D Address Format	8
Figure 5. Two Connected Subnets	9
Figure 6. IP Address Conversion	19
Figure 7. IP Special Filter Configuration	20
Figure 8. IGP and EGP	26
Figure 9. RIP, OSPF, the Routing Table and the Forwarding Engine	26
Figure 10. ISO Reference Model	37
Figure 11. OSPF Common Header Frame	38
Figure 12. OSPF LSA Header Frame	40
Figure 13. OSPF Router Link Advertisement Frame	41
Figure 14. OSPF Network Link Advertisement Frame	43
Figure 15. OSPF Summary Link Advertisement Frame	44
Figure 16. AS External Link Advertisement Frame	45
Figure 17. Sample OSPF Configuration	48
Figure 18. BGP within IP Router Code	56
Figure 19. Protocol Layering	64
Figure 20. TCP sliding window	66
Figure 21. TCP Header Format	67
Figure 22. Establishing a TCP connection	69
Figure 23. Closing a TCP connection	70
Figure 24. TCP pseudo-header	71
Figure 25. TCP Finite State Machine	73
Figure 26. IPX Packet	81
Figure 27. Best Route Determination -- Example 1	84
Figure 28. Best Route Determination -- Example 2	84
Figure 29. Best Route Determination -- Example 3	85
Figure 30. Best Route Determination -- Example 4	85
Figure 31. SPX Header	89
Figure 32. SPX Spoofing with duplicate routes with the same IPX cost	94
Figure 33. SPX Spoofing between two clouds in Default Route mode	95
Figure 34. Basic SPX Spoofing Configuration	95
Figure 35. SPX Spoofing Configuration within a few LANs	96
Figure 36. Composite IPX Network	96
Figure 37. Sample basic configuration	99
Figure 38. Two IPX RIP clouds connected via two redundant links and four exit routers	100
Figure 39. Two IPX RIP clouds connected via redundant WAN links on two exit routers	100
Figure 40. Two IPX RIP clouds connected via redundant WAN links	

on three exit routers	100
Figure 41. Three IPX RIP clouds connected improperly	101
Figure 42. Three clouds connected via default route	101
Figure 43. Branch offices connected to headquarters via default route	102
Figure 44. Branch offices connected to headquarters via default route	103
Figure 45. Duplicated network numbers in different RIP clouds.	103
Figure 46. IPX Configuration Example	106
Figure 47. IPX Router in SR or SRT Mode.	110
Figure 48. PIR Configuration Example	117
Figure 49. Central Area with Connected Satellite Areas	118
Figure 50. PIR and External Loops	119
Figure 51. Simultaneous routing and bridging	122
Figure 52. ClearSight monitors XL20-1 using address 128.100.1.1	123
Figure 53. ClearSight monitors XL20-1 using address 128.101.1.1	124
Figure 54. ClearSight monitors XL20-1 using VP's address 128.102.1.1	124
Figure 55. ClearSight monitors XL20-1 using VP's address 128.102.1.1	125
Figure 56. Routing and Bridging in parallel	127
Figure 57. Sample configuration with IPX router using virtual port.	129
Figure 58. Sample IP Routing configuration.	132
Figure 59. Sample Source Route Bridging configuration	136
Figure 60. Sample Source Route Bridging configuration	137
Figure 61. Three Typical SLCS configurations.	143
Figure 62. General PassThrough	146
Figure 63. Point-to-Point in SDLC-Specific Mode	147
Figure 64. Multidrop PassThrough in SDLC-Specific Mode	147
Figure 65. SNA Network.	148
Figure 66. Olicom LAN Multiport Network	148
Figure 67. Combined Networks Using PassThrough for SNA Traffic.	149
Figure 68. General Point-to-Point Mode	149
Figure 69. SDLC Virtual Multidrop Mode	150
Figure 70. Local Multidrop Mode	150
Figure 71. SLCS LLC2-to-SDLC Session.	153
Figure 72. SLCS SDLC-to-LLC2 Session.	154
Figure 73. SLCS Configuration with SLCS and Non-SLCS Sessions	154
Figure 74. SLCS-based Local Termination	155
Figure 75. LLC2-based Local Termination	156
Figure 76. PU 2.1 Protocol Stacks	158
Figure 77. Typical SLCS Configuration with PU 2.1 devices	158
Figure 78. PU 1.0 Devices without SLCS.	159
Figure 79. PU 1.0 devices with SLCS	159
Figure 80. Primary and Secondary SDLC Nodes.	161
Figure 81. Primary and Secondary SDLC Nodes.	162
Figure 82. Converting a Frame from SDLC to LLC	163
Figure 83. Connecting SLCS to Ports, Interfaces and Processes	164

Figure 84. Three Typical SLCS configurations	168
Figure 85. FID2 Frame Segmentation	176
Figure 86. Multidrop and Group Polling	177
Figure 87. Complex SLCS Topology	180
Figure 88. Normal bridging and DLSw	182
Figure 89. Example configuration	183
Figure 90. Example DLSw configuration	184
Figure 91. Connecting SNA devices	188
Figure 92. Connecting NetBIOS devices	191
Figure 93. Bridging and Routing	193
Figure 94. Priority values assignment to specific SAP numbers	203

Tables

Table 1. OSPF Common Header	39
Table 2. Link-State Advertisement Header	40
Table 3. Router Link Advertisement Packet	41
Table 4. Network Link Advertisement Packet	43
Table 5. Summary Link Advertisement Packet	44
Table 6. AS External Link Advertisement Packet	45
Table 7. TCP port numbers	65
Table 8. TCP Header Details	67
Table 9. States of the TCP FSM	72
Table 10. IPX Packet Header	81
Table 11. SPX Header	90
Table 12. Additional SLCS Features	152
Table 13. Key differences between LLC1 and LLC2	160
Table 14. Further comparison of SDLCand LLC2	161
Table 15. NetView Agent Commands to the Router	179
Table 16. NetView Agent Messages	179
Table 17. Types of NetView Alerts	179



1. Introduction to Routing

This chapter introduces the “Routing” section of this book (chapters 1-10 in this volume). The “Routing” section of this document consists of the variety of internetworking routing protocols supported by Olicom. For a review of the fundamental concepts of routing, refer to chapter 1, *Internetworking Principles* in volume 1 of this book. For a more detailed explanation of Olicom supported routing protocols, refer to the appropriately named chapters.

Sections

- *Internet Protocol (IP)*
- *Routing Information Protocol (RIP)*
- *Open Shortest Path First (OSPF)*
- *Border Gateway Protocol (BGP)*
- *Internetwork Packet Exchange (IPX)*
- *Protocol Independent Routing (PIR)*
- *Virtual Port*
- *ClearSession Protocol (CSP)*

IP

The Internet consists of interconnected physical networks. The Internet devices include host computers, networking devices, and internetworking devices. Each machine in the Internet is identified by a unique, 32-bit address. This address, also known as the Internet Address, identifies the location of the host or device. In addition to an address, these hosts or devices often have a name that can be mapped to their address providing an additional identification scheme.

IP is the TCP/IP session-layer protocol that regulates packet forwarding by tracking internet addresses, routing outgoing messages and recognizing incoming messages.

There are three primary classes of IP addresses used in the Internet:

- *Class A* for groups with more than 65,536 hosts and devices.
- *Class B* for groups with between 256 and 65,536 hosts and devices.
- *Class C* for groups with less than 256 hosts and devices.

► **Note:** The above classes become more and more historical; new IP routing protocols, OSPF and BGP-4, relax the addressing rules and provide for classless addressing.

For more information on IP routing, refer to chapter 2, *IP Routing* in this volume.

RIP

The Routing Information Protocol (RIP) is one protocol in a series of routing protocols based on the Bellman-Ford (or distance vector) algorithm. RIP is an “interior gateway protocol” (IGP). RIP was designed to work with moderate-size networks using reasonably homogeneous technology. Thus it is suitable as an IGP for many campuses and for regional networks using serial lines whose speeds do not vary widely. Olicom implementation of RIP is compliant with the Internet standard described in RFC 1058 and offers the following Olicom proprietary solutions:

- support for Smart Advertising. The Smart Advertising allows to reduce amount of RIP updates on WAN links - updates are transmitted until an acknowledge is received.
- piggy-back is an extension to smart advertising in that updates are sent only when the data traffic is being sent to the other side of the link
- support for unnumbered and numbered Point to Point links - a subnet mask is advertised over point-to-point links.
- support for secondary address

- route filtering based on import/export policy
- network filters for control IP RIP protocol learning and sending operation

For more information about RIP, refer to chapter 3, *RIP* in this volume.

OSPF

OSPF is a routing protocol used on TCP/IP networks that takes into account network loading and bandwidth when routing information over the network.

OSPF runs *on top of IP*. OSPF is a dynamic link-state routing protocol that routes IP packets based on information contained in the IP packet header. OSPF provides interoperability and responds quickly and dynamically to topology changes, yet involves relatively small amounts of routing protocol traffic.

The routing hierarchy of OSPF has the ability to divide a single IP class A, B, C network into many subnets of various sizes.

For more information on OSPF implementation, refer to chapter 4, *OSPF* in this volume.

BGP

The Border Gateway Protocol (BGP) is an inter-Autonomous System routing protocol. The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems. This information is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and some policy decisions at the AS level may be enforced. Olicom implementation is compliant with BGP-4 requirements, as defined in RFC#1771, which provide a new set of mechanisms for supporting classless interdomain routing. BGP-4 also introduces mechanisms which allow aggregation of routes, including aggregation of AS paths.

For more information about BGP, refer to chapter 5, *Border Gateway Protocol* in this volume.

IPX

IPX is part of Novell NetWare's protocol stack, used to transfer data between network servers and workstations. IPX packets are encapsulated and carried by packets used in Ethernet and frames in the Token Ring networks.

IPX internet packet consists of two fields, a control field (30 byte header) and a data field, that can be from 0 to 4434 bytes.

For more information on IPX routing, refer to chapter 7, *IPX Routing* in this volume.

PIR

PIR is Olicom's implementation of DSPF (Discovery Shortest Path First). DSPF automatically and dynamically chooses the best path for a packet traveling in a meshed network. In addition, DSPF is an adaptive routing protocol that periodically verifies and changes routing tables based on traffic conditions within the network.

PIR provides all of the benefits of contemporary routing protocols to protocols that do not support a native dynamic routing architecture.

For more information on PIR routing, refer to chapter 8, *Protocol Independent Routing (PIR)* in this volume.

Virtual Port

The Virtual Port is Olicom proprietary solution which provides a virtual network for the reception and transmission of frames from router applications onto the bridged network. The feature allows for simultaneous routing and bridging of the same protocol (IP/IPX) in the same XL unit. When routing on a group of physical ports is disabled they may be perceived by the IP/IPX router as one “virtual” port, and the protocol traffic may be bridged between them, and routed between this “virtual” port and ports, on which IP or IPX router is active, and protocol traffic routed. The Virtual Port on the ILAN XL is implemented as an internal port which is always present and active, and appears to the user as yet another standard port. The Virtual Port has no physical connection to physical interfaces.

For more information about Virtual Port, refer to chapter 9, *Virtual Port* in this volume.

CSP

The ClearSession Protocol (CSP) is designed to provide network redundancy and preserve active sessions if one of the network devices (IP router, or SR bridge) fails. CSP is supported for IP and for Source Route bridging.

ClearSession-configured devices monitor and provide backup services for each other. When one ClearSession device or link goes down, another attempts to take over its duties and thereby prevent data loss. See chapter 10, *ClearSession Protocol* in this volume for more.



2. IP Routing

This chapter explains Internet Protocol (IP) routing as implemented by Olicom.

Sections

- *Overview*
- *Technical Discussion*
- *Network Considerations*

Overview

Olicom routers support IP routing as an optional feature. Olicom's implementation of IP conforms to the IP standard defined by the Transmission Control Protocol/Internet Protocol (TCP/IP) suite. The implementation supports dynamic routing using the Routing Information Protocol (RIP) and the link state routing protocol using Open Shortest Path First (OSPF) as the Interior Gateways Protocols, and the Border Gateway Protocol version 4 (BGP-4) as an Exterior Gateway Protocol.

Technical Discussion

This section discusses the following IP Routing concepts:

- IP Addressing
- Secondary IP Addressing
- Masks
- Subnet Addressing
- Point-to-Point IP Addressing (numbered and unnumbered)
- IP Filters
- IP Broadcasting

IP Addressing

The IP protocol requires you to assign a unique address for each network connection. IP addresses use a 32-bit address field with bits that are numbered from 0 to 31. This 32-bit field is divided into two sections. One section, called the host number, identifies the host. The other section, the network number, identifies the network where the host resides. Therefore, hosts that are attached to the same network always use the same network number.

The following subsections provide background information to help you assign IP addresses to your Olicom router:

- Network Number Assignments
- Classes of Addresses

Network Number Assignments

To insure that Internet addresses are unique, a central authority called the Network Information Center (NIC) assigns the network number portion of the IP address to each network. Because the NIC assigns the network portion of the address, there is never a problem of having a duplicate network number in the Internet. However, it is the responsibility of the local network manager to assign unique host numbers to each node or connection on a network.

Many organizations that use the TCP/IP protocol assign network addresses on their own because they do not plan to connect to the Internet. However, even if you have no plans to join the Internet, you should contact the NIC to assign and register unique network numbers to your organization. Then if your organization decides to join the Internet, you will not need to assign new IP addresses and construct new routing tables.

Classes of Addresses (RIP only)

Classes apply to RIP only; OSPF and BGP-4 are classless routing protocols.

You can determine the class of the address from its highest-order (leading) bits. There are four classes of IP addresses. Depending on the number of hosts on the network, the NIC generally assigns either Class B or Class C network addresses. Therefore, if the network has 254 or fewer hosts, the NIC assigns a Class C network address. If a network has 255 or more hosts on a network, the NIC assigns a Class B network address.

- *Class A Address Format* -- the 32-bit address has the leading bit set to 0, a 7-bit network number, and a 24-bit host address. The 125 Class A networks can have up to 16,777,214 hosts per network.



Figure 1. Class A Address Format

- *Class B Address Format* -- the 32-bit address has the two highest-order bits set to 1-0, a 14-bit network number, and a 16-bit host address. The 16,382 Class B networks can have up to 65,534 hosts per network.

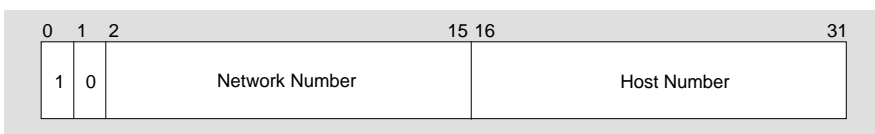


Figure 2. Class B Address Format

- *Class C Address Format* -- the 32-bit address has the three leading bits set to 1-1-0, a 21-bit network number, and an 8-bit host address. The 2,097,152 Class C networks can have up to 254 hosts per network.

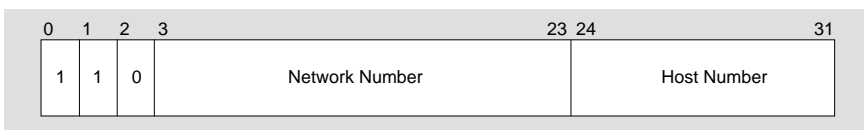


Figure 3. Class C Address Format

- *Class D Address Format* -- this 32-bit multicast address has the four leading bits set to 1-1-1-0 and a 28-bit multicast address.

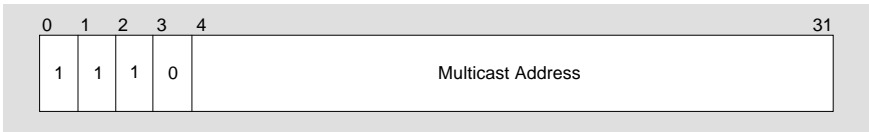


Figure 4. Class D Address Format

Secondary IP Addressing

Secondary IP addressing enables you to define multiple IP addresses for the same port. This means that in addition to the primary address of a port, a port can have additional addresses available for another purpose. You can define up to a maximum of 100 secondary addresses per router.

For example, you could assign one port 10 secondary IP addresses and have 90 secondary addresses for the remaining ports on that router. You might want to have a second IP address available for some of the following reasons:

- Some IP routing protocols such as RIP do not allow you to separate subnets that belong to the same network. To enable one of these protocols without changing your the network configuration, define secondary IP addresses to another subnet.
- Class C networks are limited to 254 hosts. If you have a Class C network that has more than 254 hosts, you could, using secondary IP addressing, divide the network into two different logical networks and have twice as many (508) hosts available
- If you need additional addresses to complete the transition between a bridge-based network and a router-based network.

► **Note:** When secondary addresses are configured on router module, all neighboring routers on same physical network **MUST** also use secondary addresses configured from same nets or subnets. Inconsistency may result in routing loops and other problems.

Subnet Addressing

Subnets are logical subdivisions of a single Internet network number. These subnet networks are interconnected by routers and each is assigned a unique network number. Subnets allow an organization to use a single Internet network number for multiple physical networks. You can use subnets with any class of Internet addressing except for Class D (multicast). It is the IP mask in conjunction with an IP address that identifies the network, subnetwork and host parts within a given address.

A subnet address, such as 135.015.001.002 (with mask configured 255.255.255.0), is created using the following NIC guidelines:

- *First Octet (135)* -- Identifies the internet network number (assigned by the NIC)
- *Second Octet (015)* -- Identifies the internet network number (assigned by the NIC)
- *Third Octet (001)* -- Identifies the subnet (assigned by the network administrator)
- *Fourth Octet (002)* -- Identifies the host (assigned by the network administrator)

For the same address, 135.15.1.2 configured with mask 255.255.0.0:

- First and Second Octets identify the network number
- Third and Fourth Octet identify host

Figure 5 shows two subnets created from the single Class B Internet address 135.015.000.000.

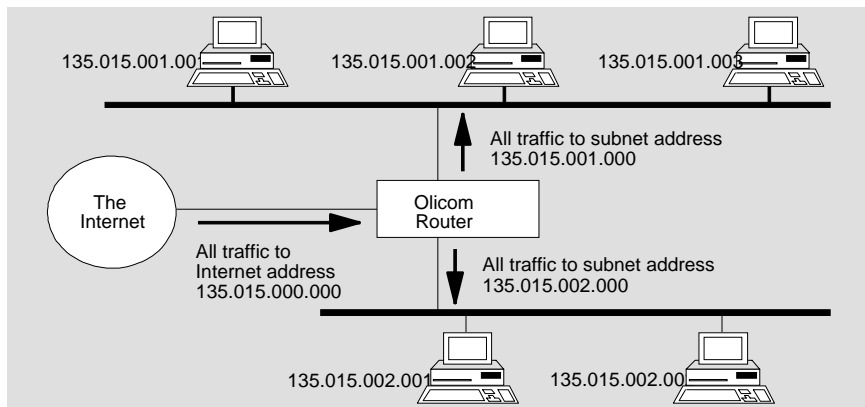


Figure 5. Two Connected Subnets

Variable-Length Subnet Addressing

IP enables you to include an address mask to each advertised destination. This mask indicates the range of addresses being described by a particular route. During IP packet transmission, the packet is always forwarded to the network that is the best match for the destination of that packet. Variable-length subnetting enables you to specify more precisely the packet destination by breaking a single IP class network number into several subnets of different sizes. The available

address space should be split up, using subnet masks, into distinct subsets of addresses that cover the original set of addresses.

► **Note:** Subnet masks must be assigned so that the best match for any IP destination is unambiguous.

For example, an IP address of 128.185.0.0 with an IP mask 255.255.0.0 actually describes a single route to destinations 128.185.0.0 - 128.185.255.255. Host routes are always advertised with a mask of 255.255.255.255, indicating the presence of only a single destination. Further, a Class B network 128.185.0.0/255.255.0.0 can be broken down into 31 variable-sized subnets: 15 subnets of size 4K, and 16 subnets of size 256 ($15 * 4K + 16 * 256 = 64K$).

The following is a sample procedure that creates variable-length subnet addresses:

1. Break the network into 16 subnets of 4K in size by using the mask of 255.255.240.0 (in hex FFFF.F0.00).
2. Divide one of the 4K subnets into 16 subnets of 256 addresses.

You should now have:

- 15 subnets of size $4096 = 61440$
- 16 subnets of size $256 = 4096$

Totalling: 65536 (the original set of addresses available in Class B network)

Examples of network addresses together with their mask:

128.185.32.0, 255.255.240.0 - subnet size is 4K addresses

128.185.224.0, 255.255.240.0 - subnet size is 4K addresses

128.185.17.0, 255.255.255.0 - subnet size is 256 addresses

128.185.30.0, 255.255.255.0 - subnet size is 256 addresses

You can further split one of the 256 subnets into smaller subnets. For example, with the mask 255.255.255.248 you could divide a 256-addresses subnet into 32 subnets of 8 addresses each.

Point-to-Point IP Addressing

Olicom enables you to use both numbered and unnumbered point-to-point IP connections. Unnumbered point-to-point connections provide additional flexibility when you are configuring your network. This flexibility is especially helpful when you are configuring WAN lines for OSPF. In addition, there are certain advantages to using unnumbered point-to-point connections when considering system management, configuration, security and other policy issues.

When you use unnumbered point-to-point connections, the address for a particular connection is configured to 0.0.0.0. Unnumbered point-to-point connections are used for the following reasons:

- As a method to preserve available IP addresses.
- To configure connections without using complex IP numbering and subnetting schemes.

Some restrictions that apply to unnumbered point-to-point connections include:

- Ping and ClearSight management communications are restricted to numbered connections.
- In OSPF with Virtual Links configured, both virtual neighbors must communicate over numbered ports.

IP Filters

Olicom's IP implementation allows you to select filters that define the following conditions:

- The source or destination address pattern and a relevant bit mask that indicate which bits from the address should be used in comparison with the actual IP address.
- Which IP protocols should be filtered.
- The restrictions for destination protocol ports of the TCP or User Datagram Protocol (UDP) datagrams.
- Whether the messages that match the filter should be rejected or accepted.
- The access to the ports of the IP device; that is the ports at which to discard or accept the matching datagrams when received from or sent to.

The device tests IP addresses against the conditions of the subsequent filters one-by-one starting with the lowest numbered filter. The decision whether to reject or accept the datagram is made upon the first match according to these rules:

 **Note:** To reject all traffic to a specific location, in addition to selecting the filtering logic, you must also set all protocol filters you could encounter.

- If the filter logic is set to reject, the datagram is discarded.
- If the filter logic is set to accept, the datagram is forwarded.
- If no filter logic is set, the datagram is forwarded.

No further tests are conducted to ensure that the datagram satisfies the conditions of remaining filters. Changing the order of the filters can affect traffic control. To

indicate the IP addresses that should be compared and those that should be ignored, set the mask bits to the following:

- Mask bits set to 1 indicate which bits from the IP address should be used in comparisons with the relevant IP address pattern defined in the filter.
- Mask bits set to 0 indicate which bits from the IP address should be ignored in comparisons with the relevant IP address pattern defined in the filter.

So, if the IP source/destination address pattern is 128.100.1.0 and the bit mask is 255.255.255.0, datagrams originating from IP network 128.100.1.0 match and should be compared.

IP Broadcasting

There are two kinds of IP broadcasts:

- **Limited broadcasts** are destined to a local network (i.e., to the same network where the originating host resides). In particular circumstances these broadcasts may be forwarded, or relayed, by routers to other IP networks. In such cases the forwarding router usually somehow modifies the packet. A good example of such behavior is the handling BOOTP requests. Each time a BOOTP request broadcast traverses a router, the BOOTP hop counter is incremented. The limited broadcast IP address is 255.255.255.255 but several older TCP/IP implementations use 0.0.0.0. Olicom's routers support both.
- **Directed broadcasts** are destined to a particular network or a group of subnets belonging to the same natural network. The destination networks can be either local or remote. An IP address consists of a network and host portion. The address for a directed broadcast has the network portion set to the destination network number and the host portion set to all ones (or all zeros in earlier implementations; Olicom supports both). For example 128.10.255.255 is a directed broadcast to the network 128.10.0.0. If this network is subnetted, the broadcast is destined to each subnet.

Limited Broadcasts

Olicom's implementation of the IP router makes it possible to disable the forwarding of limited broadcast packets or enable the forwarding of only BOOTP broadcasts or of all limited broadcasts (including BOOTP).

There are two ways to forward limited broadcasts:

- **Helper address**

When a router receives a limited broadcast on a port, it replaces the destination broadcast address (i.e., 255.255.255.255) with a helper address previously defined on this port. If the source IP address is invalid, it is replaced with the

receiving port's address. The helper address can be any valid IP address, including directed broadcast addresses. If the new destination address of the packet is a directed broadcast to a natural net then the packet is forwarded as described in *Directed Broadcasts* on page 13. Otherwise the packet is forwarded according to the routing table information. If the helper address is a directed broadcast, the packet can be delivered to more than one end station.

It is possible to have several helper addresses on a given port. In such cases, a copy of a broadcast is sent to each of these helper addresses. If helper addresses are globally enabled but there are no helper addresses defined on a port then the port is excluded from forwarding broadcasts.

- **Spanning Tree**

Spanning Tree eliminates network loops by introducing some ports of a bridge into the BLOCKING state. A port in such a state is not used to forward bridged traffic. Only ports in the FORWARDING state bridge packets. If a router receives a broadcast on a port in the FORWARDING state, the packet is flooded via all other FORWARDING ports. Broadcasts received on a port in a state other than FORWARDING are discarded.

These two methods are mutually exclusive - it is impossible to use them both simultaneously.

Directed Broadcasts

There are two kinds of IP directed broadcasts:

- **A directed broadcast to a natural network**

Any packet received by a router is checked to see whether it is a directed broadcast to a natural network: first, the class of the destination IP address is determined and next the contents of the host portion is scanned - if there are all ones or all zeros then the packet is assumed to be a broadcast and is forwarded in a special way not using the IP routing table information. Although such a broadcast is destined only to one network (if no subnetting) or to some subnets, it may traverse all possible networks.

Example:

Let's suppose we have a physical network 140.10.0.0 with mask 255.255.0.0. It is also a natural network because the address belongs to class B and the natural mask for this class is 255.255.0.0. The address 140.10.255.255 is a directed broadcast to the natural network. Now let's suppose that our network is subnetted, i.e., that there is not one physical segment but several with addresses belonging to the natural network:

140.10.1.0 with mask 255.255.255.0

140.10.2.0 with mask 255.255.255.0

140.10.3.0 with mask 255.255.255.0

.... etc.

The address 140.10.255.255 is still a broadcast address but destined to all listed networks. In both cases each router assumes that a packet with such an address is a broadcast and uses a special method to forward it.

- **A directed broadcast to a given subnet**

When a router receives broadcasts to a non-local subnet (i.e., network with IP mask longer than natural), it treats the packet as a unicast and uses routing table information to forward it. Only routers directly connected to the destination subnet can recognize that the packet is a broadcast. In such a case the router sends (broadcasts) the packet to all stations in this subnet.

Example:

Let's continue the previous example. If the network is subnetted, the address 140.10.1.255 is a directed broadcast to the subnet 140.10.1.0. Only routers directly connected to this subnet know that 140.10.1.255 is a broadcast. All other routers generally do not know about it and use the IP routing table to forward packets with such a destination address.

There are two ways to forward natural network directed broadcasts:

- **Reverse Path Forwarding**

A router forwards a copy of a received broadcast packet along all connected links if and only if the packet arrives on the port which is on the best route between this router and the source of the packet. Otherwise, the packet is discarded. If there are several equal routes to the source of the broadcast then the route which has the lowest next-hops IP address is assumed to be the best one.

This method was designed for distance-vector routing protocols like RIP. In a pure RIP environment RPF guarantees loop-free broadcasting. However with this method broadcast duplicates are likely to occur.

Using RPF in an OSPF environment may lead to broadcast loops and eventually broadcast storms. Generally, this happens when the best route from the broadcast source to a given router is different than the best route in the opposite direction.

- **Spanning Tree**

Using the Spanning Tree to forward directed broadcasts works exactly in the same way as for limited ones (see *Limited Broadcasts* on page 12). Using

Spanning Tree is independent of routing protocols, i.e., this method works well with both RIP and OSPF.

These two methods are mutually exclusive.

IP Broadcasting in a Frame-Relay Group Mode Environment

There are a few restrictions when you want to broadcast IP datagrams via Frame-Relay Group Mode ports. The first one is that these ports do not support Spanning Tree - you cannot use methods based on transparent bridge Spanning Tree.

The second restriction is that the Frame-Relay network should be fully-meshed in order to propagate broadcasts to all. Fully-meshed means here that each station has a direct virtual circuit to any other host/router in this network. In many cases, however, a configuration is not fully meshed. A star topology may be an example. In this topology a broadcast generated at the central station is delivered to all other stations but a broadcast originated in any other station is sent only to the central point. This may not be acceptable in some cases. To solve this problem we can force the central station to re-send broadcasts received on one PVC to all other PVCs belonging to the same port. In other words, this station acts as a broadcast server on the Frame-Relay network.

In these two configurations (fully-meshed and the star topology with the broadcast server) Reverse Path Forwarding will work regardless of the used routing protocol on Frame-Relay (RIP or OSPF) if there are no additional networks connecting the stations.

IP Route Import/Export Policies

An IP router can run multiple routing protocol instances. Although the domains of these routing protocols are independent in most cases, you may want to distribute routes gathered by one protocol to another. The strategy which says which routes should be redistributed (and how) is the IP router's import/export policy.

Basic Policies

A basic policy is one or more accept/reject condition which applies to IP routes. Such a policy contains the following information:

1. whether the specified route is accepted or not;
2. a specification of the route:
 - the protocol the route comes from,
 - the net range the route belongs to,
 - the next-hop of the route,
 - for RIP routes, the port the route is learned from,

- for OSPF routes, the OSPF instance identifier the route is derived from and the tag value present in the OSPF external advertisements;
 - for BGP routes, a neighboring and originating autonomous system number the route is received from and the route's origin,
3. and optionally some features which the route should receive when accepted:
- the preference the route receives when it is inserted into the routing table,
 - the metric the route is advertised with,
 - additional information when exporting non-OSPF routes into OSPF instance. This information includes metric type and OSPF tag value.

Note that policy criteria are ANDed together, so that a route must match all conditions specified in the policy for it to be a match, with the exception of the protocol that the route is derived from: when a combination of protocols is specified, a route from any of them will match that criterion.

Complex Policies

A complex policy is a list of basic policies. A router tests a given route against the conditions from the list to determine whether the route should be imported to or exported from the routing table. The tests are made one by one. The first match determines whether the router accepts or rejects the route. The order of the policies on a list is essential. By changing that order, you can change the behavior of the router completely. If none of the conditions match, the router rejects (ignores) the route.

Import and Export Policy

IP route import and export policies (available in XL 6.0 and higher) let you control which routes are imported from and exported to a given protocol (OSPF, RIP or BGP) running under IP. The words import and export are used in terms of the IP routing table: you can export from the common routing table and import to the common routing table.

An import or export policy has two essential characteristics, Scope and Applied Policies.

The scope of a policy is the range of the policy's power. You can set separate policies to export certain routes from the IP routing table to:

- all OSPF instances or a certain instance of OSPF,
- all BGP peers or a certain BGP peer,
- all RIP ports or a certain RIP port.

The same applies to import policy.

The other major characteristic of a policy is the list of basic and complex policies that is applied to it. You might, for example, define a policy that matches all routes from a certain BGP autonomous system.

Similarly to defining complex policies, the order of applied policies is essential: the first matching entry on the list determines whether or not a route is imported or exported. We may show following situations how to apply basic or complex policies:

- you can specify policies which match each route or a group of routes you want to accept. Any other routes will be rejected, so you don't have to specify them.
- if the list of accepted cases is long, or infinitive, or it is just easier to specify routes which you don't want to accept, you can create rejecting policies and place them in the beginning of the export or import list and then add a policy which is accepting all routes.
- you can use complex policies instead of consecutive list of basic policies. It is just shorter or more convenient description of the policies.

When you choose to define an export or import policy, both for the whole range and for the certain scope, you must be aware that more specific definition takes precedence. For example, consider a situation in which a definition for all BGP peers and a definition for a certain one are present. First, the list for the specific BGP peer is searched. If the route matches any policy in the list - it is accepted or rejected according to type of this policy. If there is no such match, the list for all peers is looked through.

Policies that are invalid from the viewpoint of a given protocol - for example, a RIP export policy which tries to violate the split-horizon - are ignored. Policies that do not make sense - for example, to import RIP routes from OSPF - are also ignored.

After a master reset of an XL, the default Import/Export Policy settings are as follows.

- All OSPF routes are imported to the routing table. Also all OSPF routes derived from a given OSPF instance are exported back to this instance. This is a hard-coded policy and cannot be deleted. OSPF requires this behavior to ensure a consistent view of the routing domain's topology in all routers.
- All OSPF routes are exported from the routing table to RIP.
- All RIP routes are imported to the routing table.
- All RIP routes are exported back to RIP.
- All RIP routes are exported from the routing table to all OSPF instances.

- No BGP routes are imported to the routing table.
- No BGP routes are exported to RIP and OSPF.
- All LOCAL routes are exported to RIP.

Default Policy

Sometimes the router does not know a route to some destinations. To solve this problem a default routing is introduced. Some routers can advertise the default route along the routing domain. This route appears in protocol advertisements as network 0.0.0.0 with mask 0.0.0.0. When the router receives such an update, no further configuration is required in order to forward a packet to an unknown destination. It is passed to the next-hop associated with the learned default route.

Some networks are configured by the administrator to be the ones to which traffic to unknown destinations is forwarded. The administrator defines a special policy for this purpose. If a network satisfying such a policy appears in the routing table, the router advertises the default route with a next-hop which is associated with this network.

A default route may be derived only from a routing table entry that can be used to forward packets. Some entries, like BGP aggregates, are present in the routing table only to allow routing protocol instances to advertise them. Packets cannot be forwarded according to such entries and they can not become default routes. In other words, such entries are ignored when checking the default policy.

Default routes created according to the default policy have the protocol field set to **SPECIAL**. Their next-hops are always the same as those in the original route.

Default routes derived from directly connected multi-access networks cannot be used to forward traffic because they lack next-hop information, and they are indicated as discard routes in the routing table. However, they can be used to force a router to advertise a default route only if the router is connected to a network the default route is derived from.

After a master reset of the XL there is no preset default policy.

Network Considerations

This section provides a discussion and examples of IP network topologies features:

- IP Addressing Conventions and Guidelines
- Special Filters Examples

IP Addressing Conventions and Guidelines

This subsection discusses the IP addressing conventions and guidelines established and followed by the Internet community.

Address Notation Conventions

For ease of reading, IP addresses are generally written in dotted decimal format. This format consists four decimals; a dot separates each decimal. Dotted decimal notation divides the 32-bit address into four 8-bit fields, called octets. The four octets specify the value of each field independently as a decimal number. Figure 6 shows the conversion between a 32-bit binary IP address and its dotted decimal equivalent.

Bit Pattern	10000010	00101010	00111111	00001001
Value of Each Octet	130	42	63	9
Dotted Decimal Notation	130.42.63.9			

Figure 6. IP Address Conversion

Addressing Rules

When assigning the IP address, use the following guidelines:

1. The host portion of an IP address cannot be all one bit. According to the standard, an IP address with a host portion consisting of all zeros is interpreted as *all hosts*. For example, the IP address 128.1.255.255 is interpreted as all hosts on network 128.1.
2. The network portion of an IP address cannot be all zero bits. An IP address with the network portion consisting of all zero bits is interpreted as *this network*. For example, the address 0.0.0.63 is interpreted as host 63 on *this* network.

3. The Class A network number 127 is assigned the *loopback* function. This means that a datagram sent by a higher level protocol to a network 127 address loops back in the host. Under Berkeley Software Distribution (BSD) UNIX, network sockets are used for communication with other machines and for inter-process communication. In a BSD UNIX environment, if Program A needs to communicate with Program B running on the same machine, this is accomplished using the IP network number 127. A datagram never appears on any network with a source or destination network address of 127.
4. As Figure 7 shows, segments connected by bridges have the same network numbers (for example, 128.100 and 131.2) but different host numbers. Conversely, segments connected by routers have different network and host numbers.

Special Filters Examples

The following examples show IP special filters and refer to the configuration (Figure 7). The fields used in the examples correspond to fields displayed in the IP Filter Parameters screens. Note that masks, when used with IP filters, are more like wildcard addresses: a mask need not be a series of contiguous 1s followed by a series of contiguous 0s.

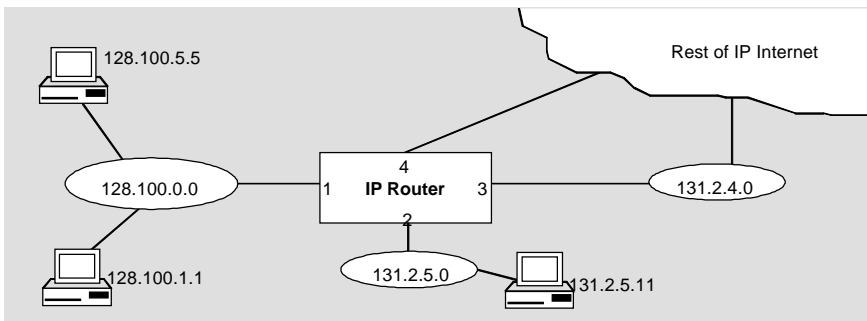


Figure 7. IP Special Filter Configuration

Example 1

This example does not allow access to selected hosts or networks using a sequence of reject filters. By default, the IP device forwards all datagrams that do not match the reject filters.

Filter	Contents	Description
Filter 1	Filter logic: Reject SRC IP Address: 128.100.0.0 SRC Mask: 255.255.0.0 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: all not selected if sent to ports: 2,3,4	All the datagrams from IP network 128.100.0.0 are discarded if forwarded to any other networks in the Internet. The datagrams (for example, SNMP messages) for the device itself are allowed.
Filter 2	FILTER 2 Filter logic: Reject SRC IP Address: 131.2.5.0 SRC Mask: 255.255.255.0 DST IP Address: 131.2.4.1 DST Mask: 255.255.255.255 Protocol filtering: TCP Port: No Relation if sent to ports: 3,4	Any TCP connections from subnet 131.2.5.0 to host 131.2.4.1 are disallowed.

Example II

This example allows the access to selected hosts or networks while the others are rejected. This is defined by using a sequence of accept filters followed by a reject filter.

Filter	Contents	Description
Filter 1	Filter logic: Accept SRC IP Address: 128.100.1.1 SRC Mask: 255.255.255.255 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: all not selected if sent to ports: 2,3,4	Allows communication of host 128.100.1.1 with any other host from the Internet.
Filter 2	Filter logic: Accept SRC IP Address: 0.0.0.0 SRC Mask: 0.0.0.0 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: TCP Port: No Relation when received from ports:1,2,3,4	Allows only TCP protocol traffic to be serviced at the device.
Filter 3	Filter logic: Accept SRC IP Address: 0.0.0.0 SRC Mask: 0.0.0.0 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: UDP Protocol Port: 161 (SNMP port) Equal when received from ports:1,2,3,4	Allows only SNMP protocol traffic to be serviced at the device.
Filter 4	FILTER 4 Filter logic: Accept SRC IP Address: 0.0.0.0 SRC Mask: 0.0.0.0 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: ICMP Port: No Relation when received from ports:1,2,3,4	Allows only Internet Control Message Protocol (ICMP) traffic to be serviced by the device.
Filter 5	Filter logic: REJECT SRC IP Address: 0.0.0.0 SRC Mask: 0.0.0.0 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: all not selected when received from ports:1,2,3,4	Rejects any datagram that has not matched the previous filters. By default, if this filter is not defined, all datagrams that do not match the filters are forwarded and no traffic control occurs.

Example III

This example excludes subsets of specified sets of datagrams from filtering. Therefore, some IP subnet traffic from specified IP networks is forwarded while other IP subnet traffic is blocked. This example is defined using a sequence of accept and reject filters that define the following:

- Subsets (subnets or hosts from a network) that use accept filters
- Sets blocked using reject filters
- Remaining traffic is forwarded by the IP device

In this example, the sequence of the filters is crucial. If Filter 1 comprised the conditions of Filter 3 no other filters would take affect because all messages would already be discarded. This would occur because the first match determines the path of the datagram.

Filter	Contents	Description
Filter 1	Filter logic: Accept SRC IP Address: 131.2.5.0 SRC Mask: 255.255.255.0 DST IP Address: 131.2.4.0 DST Mask: 255.255.255.0 Protocol: all not selected when received from port: 2	Allows the communication from subnetworks 131.2.4.0 to 131.2.5.0
Filter 2	Filter logic: Accept SRC IP Address: 131.2.4.0 SRC Mask: 255.255.255.0 DST IP Address: 131.2.5.0 DST Mask: 255.255.255.0 Protocol: all not selected when received from ports: 3,4	Allows the communication from subnetworks 131.2.5.0 to 131.2.4.0
Filter 3	Filter logic: Reject SRC IP Address: 131.2.0.0 SRC Mask: 255.255.0.0 DST IP Address: 0.0.0.0 DST Mask: 0.0.0.0 Protocol: all not selected when received from ports: 2,3	Rejects any other traffic from IP network 131.2.0.0 or its subnetworks.



3. RIP

This chapter explains Olicom's implementation of the Routing Information Protocol (IP).

Sections

- *Overview: IGP and EGPs*
- *RIP*
- *RIP Filters*

Overview: IGP and EGPs

The IP protocol is the foundation of the *Internet*, the internetwork linking corporations, universities and organizations around the globe. Each organization on the Internet generally has its own IP-based internetwork. IP meshes well with this design of independent internetworks linked into a larger internetwork.

In the terminology of IP, independent networks administered by a single authority are called *autonomous systems*. *Interior Gateway Protocols (IGPs)* are used by routers to exchange routing tables or network topology information inside an autonomous system. One or more IGPs can be implemented within an autonomous system. *Exterior Gateway Protocols (EGPs)* are used by routers to exchange routing tables between autonomous systems. Border Gateway Protocol (BGP) is the most common EGP. (Figure 8)

- ▶ **Note:** See chapter 1, *Internetworking Principles* in volume 1 for an introduction to routing tables. See the ClearSight documentation for a more detailed explanation of routing tables.

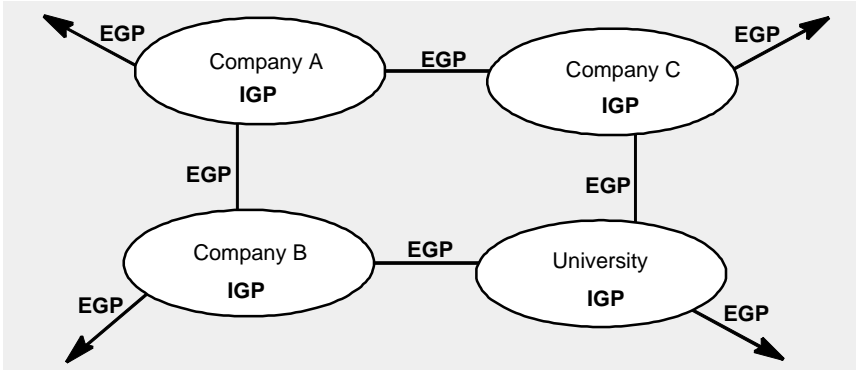


Figure 8. IGP and EGPs

RIP (Routing Information Protocol) and OSPF (Open Shortest Path First) are the two most common, standard IGPs. (See chapter 4, *OSPF* in this volume). RIP and OSPF are independent processes that build a shared routing table. The routers' *forwarding engine* uses the routing table to actually forward the packet (Figure 9).

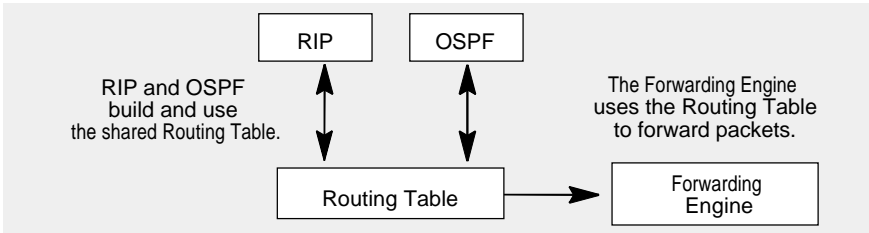


Figure 9. RIP, OSPF, the Routing Table and the Forwarding Engine

RIP

RIP provides the rules that govern how IP routers learn about routes through the network (routing tables) and exchange these routing tables with other routers. These rules are fairly simple.

Routers periodically broadcast (or *advertise*) their routing tables. Other routers hear the advertisements and add the routes to their router tables. When a router learns about a route from another router's advertisement, the route become invalid after a configurable period of time. So, if a router does not receive a rebroadcast indicating the route is valid within a configurable period of time, the router deletes the route from its routing table.

If a router receives a packet to forward and it already knows the route, it simply forwards the packet according to the route in its own routing table.

If a router receives a packet to forward and it does not know the route to the network designated in the packet's destination IP address, the packet is discarded. The router either discards the packet *silently*, or it sends a warning ICMP packet to the source station.

If a router learns of more than one path to a destination network, it uses the one with the fewest hops (the shortest distance).

- ▶ **Note:** RIP supports routing information exchange between RIP and other routing protocols, as implemented with IP import/export policy.

RIP Filters

RIP's built-in mechanisms to control the exchange of routing tables are inadequate to today's complex IP networks. XL routers support a set of filters that enable you to control how the learning and the advertising processes take place. In particular, you can use the filters to enable or disable RIP learning and advertising for a specified IP address (or groups of addresses) on specified ports.

Management Overview

You can manage the RIP filters using either ClearSight or the console commands. Both management systems use the same underlying concepts, but the console commands are used to explain the filters here. There are five console commands you use to control RIP filters. See the *Olicom XL Console Commands Online Help* for specific information on these commands.

The commands allow you to:

- Define and delete a filter using an IP Mask, an IP Address and two logical designations.
- Apply (or end the application of) the filter to a specified port or ports.

► **Note:** When a filter is applied to a port, the logical opposite of the filter is applied to all other ports, as explained below.

- Display a list of all defined filters.

The filters are defined and applied with the following terms and values:

- *IP Address:* used with the IP Mask (next item) to determine whether the filter applies to individual RIP packets. Only one filter can be defined for each IP address.
- *IP Mask:* Used with the IP Address to determine whether the filter applies to individual RIP packets. A filter is like an address wildcard: any given bit can be set to 0 or 1, independent of the other bits, as needed to control traffic with the filter.
- *Port Number:* specifies the port you want to apply the filter to.

► **Note:** When you specify a port to apply a filter to, the logical opposite is applied to all other ports. (See next item, "Accept or Reject Logic".)

- *Accept or Reject Logic:* specifies whether advertising and learning on the address(es) designated by the IP Address and the IP Mask is enabled (Accept) or disabled (Reject). When a filter is applied to a port or a set of ports, the opposite Accept/Reject logic is applied to all other ports.

- *In or Out logic*: specifies whether incoming RIP advertisements (In) or outgoing RIP advertisements (Out) are going to be subject to the filter.

Defining a Filter

To define a filter, you must first understand how a particular filter's IP Mask and its IP Address are used to determine whether the filter applies to individual RIP packets. The second part of filter definition is configuring the filter's Accept|Reject Logic. Then, the filter is applied to a particular port. When a filter is applied to a port, the same filter is applied with the opposite Accept|Reject logic to all other ports.

Using the IP Mask and the IP Address to Specify Filter Application

There are several steps of analysis that occur that allow the router to decide whether to apply a filter to a RIP packet, and then to determine which filter to apply if more than one are eligible.

1. The IP Mask of each filter is applied to the IP address in the RIP packet. The result is a logical multiplication. That is, there is a digit-by-digit comparison of every binary digit. If the digit in the IP mask and the digit in the filter's IP Address both are *1*, then the digit in the result is *1*. This is shown here with two four-digit binary numbers and the result.

	Decimal	Binary			
Number 1	5	0	1	0	1
Number 2	12	1	1	0	0
Result	4	0	1	0	0



Note: While IP Addresses and IP Masks are often shown in decimal notation, it is easier to analyze their binary equivalents.

2. The IP Mask for each filter is compared in exactly the same manner with the IP Address that you set for the filter. Again, there is a digit-by-digit comparison. If the digit in the IP Mask and the digit in the IP address in the RIP packet both are *1*, then the result is *1*.
3. If the result of the first binary multiplication (Step 1) equals the result of the second binary multiplication (Step 2), the filter is a candidate for application. The filter is not automatically applied; rather, the filter with the most specific mask is applied. (See step 4.)
4. When more than one filter results in a match, the filter with the more specific mask is applied. *More specific* means the filter with the greater number of binary digits with a value of *1*. For instance, this mask: 255.255.0.0

(binary: 11111111.11111111.0.0), is more specific than this mask: 255.0.0.0
(binary: 11111111.0.0.0).

Here are three examples of how IP Address and IP Masks are used to determine whether a particular filter is applied to individual packets.

Example 1

Address	Decimal	Binary	Results Match?
RIP Packet	128.100.105.101	10000000.01100100.01101001.01100101	Result 1 = Result 2
IP Mask	255.0.0.0	11111111.00000000.00000000.00000000	
Result 1		10000000.00000000.00000000.00000000	
IP Address	128.128.100.110	10000000.10000000.01100100.01101110	
IP Mask	255.0.0.0	11111111.00000000.00000000.00000000	
Result 2		10000000.00000000.00000000.00000000	

- The results of the two comparisons are exactly equal. Therefore, the filter is applied if no other filter with a more specific IP Mask matches.
- Usually an IP Mask is established with decimal numbers of 255 (binary: 11111111) in one or more of the four IP address positions.
- This example shows that a mask of 255.0.0.0 means that only the first number in the RIP packet's IP address and in the filter's IP Address is examined when the decision of whether to apply the filter is made.

Example 2

	Decimal	Binary	Results Match?
RIP Packet	128.100.105.101	10000000.01100100.01101001.01100101	Result1 <> Result 2
IP Mask	255.255.0.0	11111111.11111111.00000000.00000000	
Result 1		10000000.01100100.00000000.00000000	
IP Address	128.101.100.110	10000000.01100101.01100100.01101110	
IP Mask	255.255.0.0	11111111.11111111.00000000.00000000	
Result 2		10000000.01100101.00000000.00000000	

- The results of the two comparisons are not equal. Therefore, the filter is not applied to this packet.

Example 3

	Decimal	Binary	Results Match?
RIP Packet	128.100.105.101	10000000.01100100.01101001.01100101	Result 1 = Result 2
IP Mask	255.255.0.0	11111111.11111111.00000000.00000000	
Result 1		10000000.01100100.00000000.00000000	
IP Address	128.100.100.110	10000000.01100100.01100100.01101110	
IP Mask	255.255.0.0	11111111.11111111.00000000.00000000	
Result 2		10000000.01100100.00000000.00000000	

- The results of the two comparisons are exactly equal. Therefore, the filter is applied if no other filter with a more specific IP Mask matches.
- This example shows that a mask of 255.255.0.0 means that only the first and second numbers in the RIP packet's IP address and the Filter IP Address are examined when the decision of whether to apply the filter is made.

Example 4

	Decimal	Binary	Results Match?
RIP Packet	128.100.105.101	10000000.01100100.01101001.01100101	Result 1 = Result 2
IP Mask	255.255.255.0	11111111.11111111.11111111.00000000	
Result 1		10000000.01100100.01101001.00000000	
IP Address	128.100.105.110	10000000.01100100.01101001.01101110	
IP Mask	255.255.255.0	11111111.11111111.11111111.00000000	
Result 2		10000000.01100100.01101001.00000000	

- The results of the two comparisons are exactly equal. Therefore, the filter is applied if no other filter with a more specific IP Mask matches.
- This example shows that a mask of 255.255.255.0 means that only the first, second, and third numbers in the RIP packet's IP address and the Filter IP Address are examined when the decision of whether to apply the filter is made.
- If the filter shown in Example 2 and the filter shown in Example 3 are both defined, only the filter in Example 3 is applied because it has a more specific IP Mask.

Using Accept and Reject

When you define a filter, you must specify the Accept/Reject logic the filter uses. When the Accept logic is specified, RIP advertising and learning is enabled. When the Reject logic is specified, RIP advertising and learning is disabled.

► **Note:** The default condition for advertising and learning is *Accept* (enabled), so no accept filters need to be defined and applied to allow the router to use RIP.

Example Filter Definition

```
SYNTAX: IPRIP FILTERS ADD <IP ADDRESS> <IP MASK>
{ACCEPT|REJECT}
```

```
EXAMPLE: IPRIP FILTERS ADD 128.100.000.000
255.255.000.000 ACCEPT
```

This definition creates a filter that enables learning from and advertising to all IP addresses that start with 128.100.

Applying the Filter

After you have defined a filter, you must apply it to a port or a set of ports and specify whether the logic is to apply to outgoing advertisements or incoming advertisements.

Designate the Filter with the IP Address

To apply a defined filter, you must designate which of the defined filters you want to apply. You designate a defined filter with the IP address named in the filter definition. Because only one filter can be defined for any IP address, designating the IP address designates the filter.

Specify the In/Out Logic and the Port

When you apply a filter, you must specify the IN|OUT logic and the Port Number. If the In|Out logic is set to *In*, only incoming advertisements are subject to the filter. If the logic is set to *OUT*, only outgoing advertisements are subject to the filter.

The Port Number simply designates the port you want to apply the filter to. The filter is applied to all other ports using the opposite ACCEPT|REJECT logic. That is, if you apply a filter with *accept* logic to a particular port, the filter is applied to all other ports with *reject* logic.

Example Filter Application

```
SYNTAX: IPRIP FILTERS {APPLY|REMOVE} {IN|OUT} <IP  
ADDRESS> <PORT NUMBER>
```

```
EXAMPLE: IPRIP FILTERS APPLY IN 128.100.000.000 5
```

This applies the filters defined above for IP addresses 128.100.000.000 to ports. As a result of this application, the router learns advertisements for all IP addresses that start with 128.100 on port number 5.

► **Note:** Such advertisements are rejected if received on other ports.



4. OSPF

This chapter explains Olicom's implementation of the Open Shortest Path First (OSPF) link-state routing protocol.

Sections

- *Overview*
- *Technical Discussion*
- *Network Considerations*

Overview

OSPF is the Interior Gateway Protocol (IGP) specified by the Internet Engineering Task Force (IETF). OSPF runs *on top of* the Internet Protocol (IP). The OSPF protocol is a dynamic link-state routing protocol that routes IP packets based on information contained in the IP packet header. OSPF detects topological changes and immediately floods this information to the connected, participating routers. The protocol then calculates new loop-free routes.

The OSPF protocol provides interoperability and responds quickly and dynamically to topology changes, yet involves relatively small amounts of routing protocol traffic. OSPF, compared to other protocols, such as the Routing Information Protocol (RIP), reduces the amount of required traffic by forwarding only incremental changes to the routing table rather than the entire table.

The four level routing hierarchy of OSPF also provides a more flexible routing metric than other IP protocols and has the ability to divide a single IP class A, B, C network into many subnets of various sizes.

OSPF supports routing information exchange between OSPF and other routing protocols, as implemented with IP import/export policy.

Supported Features

OSPF supports the following features:

- Link state protocol that provides a basis for a loop-free algorithm
- Designated Router concept that reduces the amount of required traffic
- Minimum of routing protocol traffic (incremental updates are of changes only)

- Flexible routing metric by combining issues such as cost, speed, and distance
- Area routing that provides an additional level of routing protection within an area from all information external to the area
- Reliable flooding mechanism (protocol packets must be explicitly acknowledged)
- Four level routing hierarchy that enables multiple levels of routing protection and simplified routing management in *autonomous systems* (AS)
- Manual configuration of virtual links
- Authentication of routing protocol exchanges
- Creation of stub areas
- Host routes
- Unnumbered point-to-point interfaces

Olicom's implementation of OSPF supports the following features:

- Variable length subnetting (the ability to divide a single IP class A, B, C network into many subnets of various sizes)
- Secondary IP addresses
- Routing information exchange as implemented with IP Import/Export policy
- Split traffic on up to three equal cost routes

Technical Discussion

This section discusses the following topics:

- OSPF Protocol Relationships
- Link State Protocol
- Routing Areas
- OSPF Packet Encoding
- Link State Advertisement Packets
- Authentication Types

OSPF Protocol Relationships

Within the frame work of the International Standards Organization (ISO) seven layer reference model, OSPF and its related protocols can be found in the Network (Layer 3) and Transport (Layer 4) areas of the model. Figure 10 shows you where the OSPF protocol fits into the ISO model.

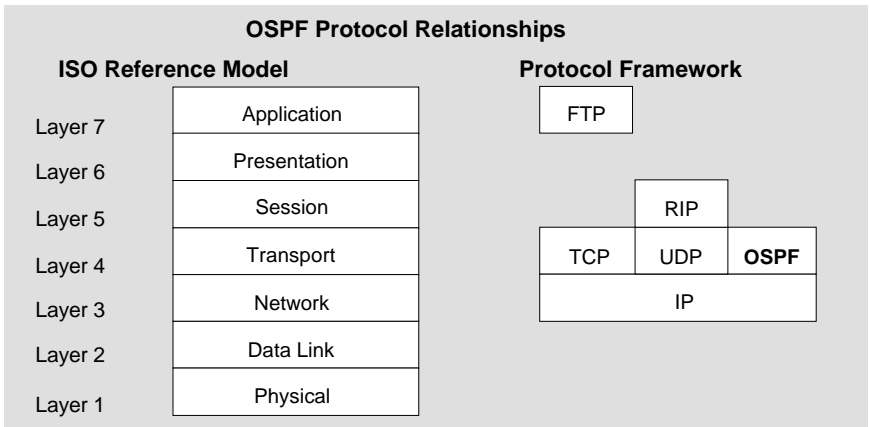


Figure 10. ISO Reference Model

Link State Protocol

OSPF implements a link state protocol. This protocol requires that each router within a network have synchronized databases. Using this common set of routing information, each router builds the shortest path tree from itself to each destination, using itself as the root. Link state protocols are unique from other protocols in that they exchange information about their links (together with the information about the state of their links) rather than routes themselves. And while

some protocols flood the entire routing table every n seconds regardless of whether there has been a change to the topology, OSPF routers flood topology information only in the event of a change to the configuration. As a result, since only changes are updated in OSPF, network routing traffic is significantly reduced.

Routing Areas

Routing areas consist of groups of networks, hosts and routers that have an interface to the area. Each area maintains its own routing algorithm and shares identical topological information with all other routers on the network.

OSPF Packet Encoding

Olicom's implementation of OSPF has compact encoding that results in faster processing of packets. OSPF has no variable-length fields in the protocol packets and makes no provisions for adding fields (that are ignored by previous-version routers).

OSPF packets are distinguished from other IP packet types (such as UDP and TCP) by the protocol field (set to 89 for OSPF) in the IP packet header.

This section describes the following OSPF packet types:

- Hello
- Database Description
- Link-State Request
- Link-State Update
- Link-State Acknowledgement

Common Header

All OSPF packets start with a 24-byte header called the common header. The common header frame format is shown in Figure 11.

Version # 1 byte	Packet Type 1 byte	Packet Length 2 bytes	Router ID 4 bytes	Area ID 4 bytes	Checksum 2 bytes	Authentication Type 2 bytes	Authentication ID 8 bytes
---------------------	--------------------------	-----------------------------	-------------------------	-----------------------	---------------------	-----------------------------------	---------------------------------

Figure 11. OSPF Common Header Frame

Field	Feature/Option	Number of Bytes
Version #	The OSPF version number assigned by the OSPF working group of the IETF. Olicom's implementation supports version 2 of the OSPF protocol.	1
Packet Type	One of the following OSPF packet types: <ul style="list-style-type: none"> •1 = Hello •2 = Database Description •3 = Link State Request •4 = Link State Update •5 = Link State Acknowledgement 	1
Packet Length	The number of bytes in the entire OSPF packet including the standard 24-byte OSPF packet header.	2
Router ID	The identity of the router that generates the packet.	4
Area ID	The identity of the origination area of the OSPF packet.	4
Checksum	This value is calculated using the standard 16-bit checksum algorithm for IP data packets.	2
Authentication Type	The protection scheme used by the OSPF protocol: <ul style="list-style-type: none"> •AuType 0 = no authentication •AuType 1 = simple password 	2
Authentication Data	A 64-bit field for use by the authentication scheme.	8

Table 1. OSPF Common Header

Link-State Advertisement Packets

OSPF uses the information in link-state advertisement packets to manage the network topology database. This section describes the link-state advertisement header and the following LSA packet types:

- *Router Link Advertisement*
- *Network Link Advertisement*
- *Summary Link Advertisement (for reachable IP networks)*
- *Summary Link Advertisement (for reachable AS boundary routers)*
- *AS External Link Advertisement (for reachable IP networks external to the AS)*

Link-State Advertisement Header

The link-state advertisement (LSA) header is a 20-byte frame that starts all LSA packets. The LSA header format is shown in Figure 12.

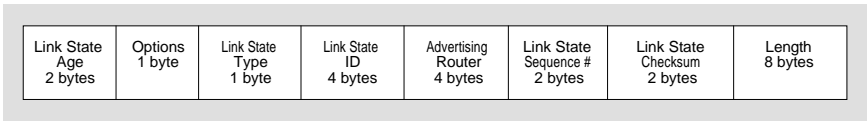


Figure 12. OSPF LSA Header Frame

Field	Feature/Option	Number of Bytes
LS Age	Estimate of the number of seconds since the last LSA was originated.	2
Options	Carries the E bit (when reset - meaning that the router presumes the area is a “stub” area) and the T bit (when set - meaning that the router can handle multiple types of service).	1
LS type	One of the following: <ul style="list-style-type: none"> • 1 = router link • 2 = network link • 3 = summary link (reachable IP networks) • 4 = summary link (reachable AS boundary routers) • 5 = AS externally reachable IP networks 	1

Table 2. Link-State Advertisement Header

Link State ID	This is dependent on the LSA packet type defined: <ul style="list-style-type: none"> • For LSA Type 1 -- the ID of the router that originated the LSA. • For LSA Type 2 -- the IP interface address of the Designated Router on the LAN. • For LSA Type 3 -- the destination network's IP address. • For LSA Type 4 -- the ID of the AS boundary router. • For LSA Type 5 -- the IP address of the reported IP destination. 	4
Advertising Router	The 32-bit ID of the router that originated the LSA.	4
LS Sequence Number	The 32-bit sequence number of the LSA.	4
LS Checksum	The Fletcher checksum of the complete contents of the LSA excluding the age.	2
Length	The number of octets in the LSA (includes the 20 byte LSA header).	2

Table 2. Link-State Advertisement Header

Router Link Advertisement

Router link advertisements (Type 1 LSA packets) are generated by a router and provide information about neighboring routers and attached LANs. Router link advertisement packets are flooded within the area only. The router link advertisement frame format is shown in Figure 13.

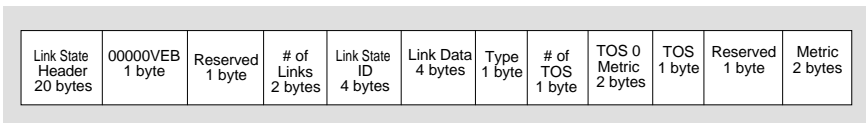


Figure 13. OSPF Router Link Advertisement Frame

Field	Feature/Option	Number of Bytes
Link State Header	The 20-byte packet that starts all Link-State Advertisement headers.	20

Table 3. Router Link Advertisement Packet

00000VEB	Bit V, when set, the router is an endpoint of an active virtual link that is using the described area as a transit area. Bit E (External), when set, indicates that the router is an AS boundary router. Bit B (Border), when set, indicates that the router is an area border router.	1
Reserved	Not user specified	1
Number of Links	The total number of links in the area of the router that generated the LSA.	2
Link ID	Identifies the entity connected to the router. The Link ID is one of the following: <ul style="list-style-type: none"> • <i>Link Type 1</i>: ID of neighbor router • <i>Link Type 2</i>: IP Address of the Designated Router on the LAN. • <i>Link Type 3</i>: IP network/subnet number. • <i>Link Type 4</i>: ID of neighbor router. 	4
Link Data	For connections to stub networks it specifies the network's IP address mask. For unnumbered point-to-point connections, it specifies the interface's MIB-II ifIndex value. For other link types it specifies the router's associated IP interface address.	4
Type	One of the following link types: <ul style="list-style-type: none"> • 1 = point-to-point link to another router • 2 = connection to <i>transit</i> network. • 3 = connection to <i>STUB</i> network. • 4 = virtual link 	1
Number of TOS	Specifies the number of costs reported for the link, not counting TOS 0, which is required.	1
TOS 0 Metric	Cost of using this router link for Type of Service 0.	2
TOS	Type of service.	1
Reserved	Not user-specified.	1
Metric	Advertised cost of the link (for traffic of the specified TOS).	2

Table 3. Router Link Advertisement Packet

Network Link Advertisement

Network link advertisements (Type 2 LSA packets) are generated by the Designated Router from all routers attached to a multi-access network and lists all the routers on the multi-access network. Network link advertisement packets are flooded within the area only. The network link advertisement frame format is shown in Figure 14.

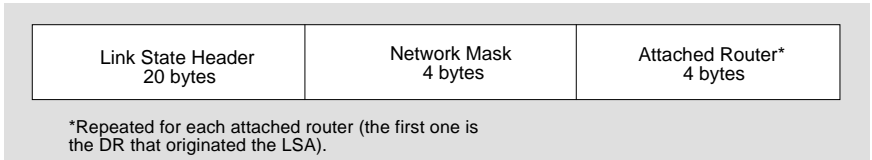


Figure 14. OSPF Network Link Advertisement Frame

Field	Feature/Option	Number of Bytes
Link State Header	The 20-octet packet that starts all Link State Advertisement headers.	20
Network Mask	The mask for the multi-access network's IP address.	4
Attached Router	The ID of each of the routers on the LAN that are fully adjacent neighbors of the Designated Router. The number of routers can be deduced from the LSA header length field.	4

Table 4. Network Link Advertisement Packet

Summary Link Advertisement

Network summary link advertisements (Type 3 LSA packets) are generated by an area border router to provide reachable IP networks outside of the area but within the Autonomous System (AS). Summary link advertisement packets are flooded within the area.

- **Note:** Each Type 3 LSA has a single IP destination. Therefore, area border routers generate a separate summary link advertisement for each reachable IP destination outside of the area that is within the AS. In addition, area border routers have separate Type 3 LSAs for each attached area.

AS boundary routers summary link advertisements (Type 4 LSA packets) are generated by an area border router to provide next hop and cost of the path information from that router to the AS boundary router.

- **Note:** Each Type 4 LSA has a single IP destination. Therefore, area border routers generate a separate summary link advertisement for each reachable AS boundary router which belongs to AS, yet is outside the area.

The summary link advertisement frame format is shown in Figure 15.

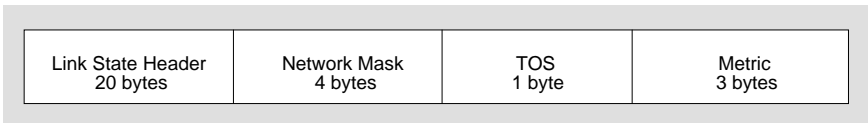


Figure 15. OSPF Summary Link Advertisement Frame

Field	Feature/Option	Number of Bytes
Link State Header	The 20-octet packet that starts all Link State Advertisement Headers.	20
Network Mask	The mask for the network's IP Address for Type 3 LSA. (For Type 4 LSA this field is not meaningful and must be zero.)	4
TOS	Type of service.	1
Metric	Advertised cost to the destination.	3

Table 5. Summary Link Advertisement Packet

AS External Link Advertisement

AS external link advertisements (Type 5 LSA packets) are generated by the AS boundary router to provide information about destinations known to the router which are external to the AS. The AS external link advertisement frame format is shown in Figure 16.

► **Note:** Each Type 5 LSA describes a single IP destination. Therefore, AS boundary routers generate a separate advertisement for each known external destination.

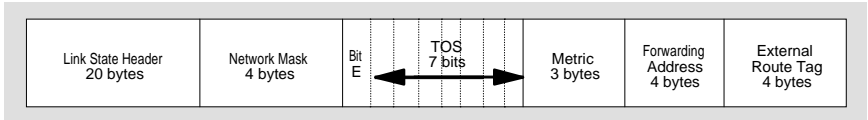


Figure 16. AS External Link Advertisement Frame

Field	Feature/Option	Number of Bytes
Link State Header	The 20-octet packet that starts all Link State Advertisement Headers.	20
Network Mask	The mask for the network's IP Address.	4
Bit E	Type of external metric: <ul style="list-style-type: none"> •One (Set) =Type 2 and the metric is considered larger than any link state path. •Zero = Type 1 and the metric is directly comparable (without translation) to the link state metric. 	1 bit
TOS	Type of service.	7 bits
Metric	Advertised cost to the destination.	3
Forwarding Address	Optimizes the final hop. The advertising ASBR places the address of the router on the most optimal path into the forwarding address field. This indicates the IP destination address where packets should be sent. If the Forwarding Address is set to 0.0.0.0 data traffic will be forwarded to the advertisement's originator.	4
External Route Tag	Open field that provides interdomain information and is not used by OSPF itself.	4

Table 6. AS External Link Advertisement Packet

Authentication Types

All OSPF protocol exchanges have an authentication type field and 64 bits of data for use by the appropriate authentication scheme. The authentication type is configured on a per-area basis. Additional authentication data is configurable on a per interface basis. For example, if an area uses a simple password scheme for authentication, you must configure a separate password for each network in the area.

The following are the authentication types supported by the Olicom implementation of the OSPF protocol:

- *AuType 0: No authentication* -- This authentication type specifies that routing exchanges in the area are not authenticated. The 64-bit field in the OSPF header can contain anything because it is not examined on packet reception.
- *AuType 1: Simple password* -- This authentication type means the 64-bit field is configured on a per-network basis. All packets sent on a particular network must contain this value in their OSPF header 64-bit authentication field. This essentially serves as a “clear” 64-bit password. When using this authentication type, you must configure the passwords of the attached networks before routers can participate in the routing domain. This guards against routers inadvertently joining the area.

Network Considerations

This section discusses the issues you should consider when designing a network that supports the OSPF protocol:

- The OSPF Autonomous System
- Hierarchical Architecture of OSPF
- Defining Supported OSPF Network Topologies

The OSPF Autonomous System

In OSPF, all areas of a network together form the Autonomous System (AS), which is a self-contained internetwork. Typically, an organization runs a network consisting of a single AS, though a large, international organization may run several ASs, one for each continent where they do business. Communications within an AS are internal. Communications between ASs are external.

Hierarchical Architecture of OSPF

A hierarchical architecture provides four levels of network structure. The routers in a hierarchical network are classified according to the level at which they function. In a four level architecture, routers are grouped into two classes known as inter and intra area routers.

Area routers are used primarily to route data within a portion of the network referred to as an area. This is called intra-area routing. Area routers also route data toward destinations in other areas. The most common attribute used to define an area of the network is geographical proximity. Backbone routers, on the other hand, are higher level routers used to transmit data between areas. This is called inter-area routing. Area routers pass information to backbone routers for inter-area routing.

It is important to understand that all the backbone routers together form an area, known as the backbone area. Backbone routers do not need to be physically connected to the backbone area, but can be connected by virtual links. All areas of a network, including the backbone area, form an organization's entire Internet network.

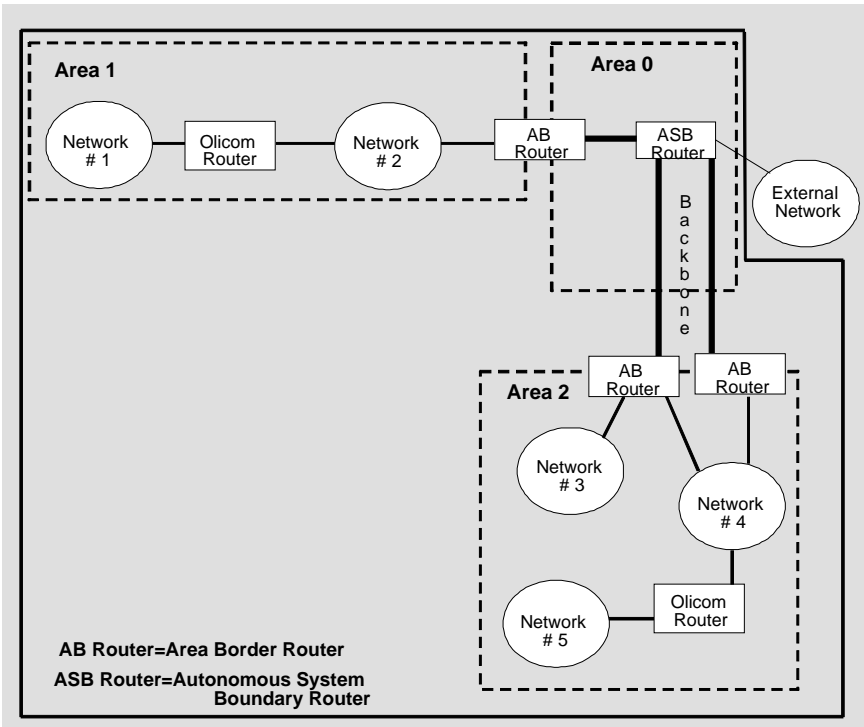


Figure 17. Sample OSPF Configuration

Defining OSPF Supported Network Topologies

The AS's topological database describes a graph that consists of routers and networks. The OSPF protocol supports two types of networks:

- *Point-to-point* -- This network type joins a single pair of routers.
- *Multi-access Networks* -- This network type supports the attachment of more than two routers.

Additionally, there are two kinds of multi-access networks:

- *Broadcast* -- This is a network that supports more than two routers and can address a single message or datagram to all attached router.
- *Non-broadcast* -- This is a network that supports more than two routers and can address a single message or datagram to each attached router, but has no broadcast capability.

Configuring OSPF


Preliminary Configuration

Before you configure OSPF on your network, complete the following:

- Create an IP addressing scheme for OSPF (for additional information on IP addressing refer to the IP Routing section of this document).
- Establish the number and type of areas of your network (including stub areas).
- Select your net ranges (as a means of aggregation).
- Establish the number of interface or ports you want (including whether to use numbered or unnumbered Point-to-Point lines.)

Configuring CFG

To enable OSPF use the following procedure to configure CFG:

 **Note:** To avoid unnecessary routing information exchange, disable IP until you have configured OSPF.

1. Assign IP addresses to interface/ports.
2. Create OSPF instance using the Router ID.
3. Create and enable OSPF areas using the Area ID.
4. Assign and enable interface/ports to OSPF areas.
5. Enable OSPF protocol globally.

OSPF Terminology

This section defines terms that apply specifically to the OSPF link-state routing protocol.

Adjacency

The relationship between selected neighboring routers for the purpose of exchanging routing information with other neighboring routers. OSPF requires that only adjacent routers synchronize their topological databases. Not every pair of neighboring routers is adjacent. Routers connected by point-to-point networks and virtual links always become adjacent. On multi-access networks, all routers become adjacent to both the Designated Router and the Backup Designated Router.

Area Border Router

A router that attaches to multiple areas. Area border routers run multiple copies of the basic algorithm, one copy for each attached area and an additional copy for the backbone. Area border routers condense the topological information of their attached areas for distribution to the backbone. The backbone then distributes the information to other areas.

Autonomous System (AS)

A group of routers that have a single coherent interior routing plan even when multiple IGPs and metrics are used to exchange routing information.

AS Boundary Router

A router that exchanges routing information with routers from more than one Autonomous System (AS). AS boundary routers have AS external routers that are advertised throughout the AS. The path to each AS boundary router is known to every router in the AS.

Backbone Router

A router that has an interface to the backbone. This includes all routers that interface to more than one area. However, backbone routers do not have to be area border routers. Routers with all interfaces connected to the backbone are considered internal routers.

Designated Router (DR)

Designated Routers generate link-state advertisements for the multi-access network, thus reducing the number of required adjacencies, the amount of routing protocol traffic, and the size of the topological database on the network. The Hello protocol selects a Designated Router for each multi-access network with at least two attached routers.

Exterior Gateway Protocol (EGP)

A class of the IP routing protocol used between Autonomous Systems. Synonyms for this term include interdomain or inter-AS.

Interface

The connection between a router and one of its attached networks. Within OSPF, both interfaces and ports can either be physical or logical in nature. An interface has state information that is obtained from OSPF and other lower level protocols. An interface to a network has a single IP address and mask.

Interior Gateway Protocol (IGP)

The routing protocol used by routers that belong to one AS. Separate AS's may run different IGPs.

Internal Router

A router that has all of its interfaces directly connected to the same area, such as a router with only backbone interfaces. Internal routers run a single copy of the basic routing algorithm.

Hello Protocol

The portion of the OSPF protocol that establishes and maintains neighbor relationships. On a multi-access network, the Hello protocol can also dynamically discover neighboring routers.

Link-State Advertisement

Describes the local state of a router or network, including the state of the router's interfaces and adjacencies. Each link-state advertisement is flooded throughout a particular area, except for ASE LSA's which are flooded throughout the whole routing domain. There are five types of link-state advertisements and the routers that issue them along with their networks form the OSPF topological database.

Link-State Algorithm

An algorithm that determines the optimal paths through a network that has a hierarchical architecture with at least two levels. The routers in a hierarchical network are classified according to function. Routers are grouped into four classes: internal, area border, backbone and AS boundary routers. Each participating router has an identical database. Each routers' local state includes the routers' usable interfaces and its reachable neighbors. All routers run the exact same algorithm. From this database, each router constructs a tree of shortest paths using itself as the root.

Lower-level Protocols

Underlying network access protocols that provide services to IP and OSPF protocols. Examples of these protocols are the X.25 packet and frame levels for X.25 private data networks (PDNs) and the data link level of Ethernet networks.

Multi-access Networks

Physical networks that support the attachment of multiple (more than two) routers. Each pair of routers on such a network is assumed to be able to communicate directly.

Neighboring Routers

Routers that share a common network or are connected through a serial link. Neighbors on multi-access networks use OSPF's Hello protocol to dynamically discover neighbors.

Network

A physical Local Area Network (LAN) or Wide Area Network (WAN).

Network Mask

A 32-bit number indicating the range of IP addresses on a single IP network. For example, the network mask for a Class C IP network is displayed as 0xFFFFFFFF00 or 255.255.255.0.

Router ID

A 32-bit number assigned to each router running the OSPF protocol to uniquely identify the router within the AS.

Routing Domain

A group of routers that have a single coherent interior routing plan even when multiple IGPs and metrics are used to exchange routing information.

Shortest Path First (SPF)

Based on a loop-free Dijkstra algorithm that computes the optimal path to all destinations in the AS from the link-state advertisement database for each area. This algorithm provides this path information to nodes within the AS only after the databases for each area have been synchronized. This method is much more efficient than RIP, since only the changes to the path information is propagated to the nodes.

Stub Area

A configuration option that selects the optimal exit point within a stub network and disables flooding of ASE LSA's information throughout the area. Routing to AS external destinations is based on a per area default only. Stub areas do not support virtual links.

Stub Network

A network that sends and receives packets, but does not allow pass through traffic.

Transit Area

A non-backbone area the two routers have in common. They are the only non-backbone areas that can carry data traffic that neither originates nor terminates in the area itself. Transit areas can support one or more virtual links.

Transit Network

A multi-access network that has more than one attached router that passes traffic between networks in addition to carrying traffic for its own hosts. In order to have a transit network, at least two network paths must be configured.

Virtual Links

A configurable routing option that enables you to restore or increase connectivity to the backbone. These links appear as numbered point-to-point or broadcast links between two area border routers.



5. Border Gateway Protocol

This chapter discusses the Border Gateway Protocol Version 4 (BGP-4) as implemented by Olicom.

Sections

- *Overview*
- *Technical Discussion*

Overview

BGP is an interautonomous system routing protocol that accumulates network reachability information about routes from packets as they traverse the network. Route attributes such as the cost or security of a path are also added. BGP reduces the bandwidth needed to exchange routing information because the information is exchanged incrementally, rather than by sending the entire database. BGP was designed to replace the Exterior Gateway Protocol (EGP). For an overview of EGP, see chapter 3, *RIP* in this volume. As the successor, BGP offers several significant advantages over EGP.

- BGP can operate with networks that have looped topologies; it uses algorithms that trim the loops out of the topology.
- BGP does not have the “count-to-infinity” problem found in many route discovery protocols because it advertises all autonomous systems on the path to a destination address.
- The full advertising capability of BGP allows a node that receives more than one possible path to a destination to choose, without ambiguity, the best path.

As implemented by Olicom, BGP-4 is compliant with the Internet standard described in RFC 1771 and offers the following features:

- The capacity to exchange routing information with RFC 1771/2 compliant routers in other autonomous systems.
- The capacity to exchange routing information with other RFC 1771/2 compliant routers within the same autonomous system.
- Route aggregation and Classless Inter-Domain Routing (CIDR).

The BGP is managed through the XL console and SNMP (ClearSight). The SNMP agent supports the standard BGP-4 Management Information Base (MIB) and Olicom's BGP private MIB.

Technical Discussion

BGP distributes routing information among multiple autonomous systems (ASs). As the name suggests, the BGP routers are located on the borders of the ASs. Within each autonomous system however, the Internal Gateway Protocol (IGP) is used to distribute routing information rather than BGP. Open Shortest Path First (OSPF) and Routing Information Protocol (RIP), for example, belong to IGP.

► **Note:** Special care must be taken to ensure consistency between BGP and IGP, as changes are likely to be propagated at different rates across the AS. There can be a time lag between the moment when a BGP router receives new routing information from another BGP router within the system and the moment the IGP is able to carry traffic to each border router. During this time lag, either incorrect routing or *black holes* can occur. If OSPF is used as the IGP, define a peer with the IGP option to allow interior routing to converge on the proper exit gateway before advertising routes through that gateway to other autonomous systems.

BGP uses the Transmission Control Protocol (TCP) as its transport layer protocol. All packets to and from the BGP are carried by the TCP subsystem. Potential problems such as reliable receipt of traffic, segmentation, and so on are handled by this transport layer. Figure 18 shows the relationship of BGP and the TCP subsystem to the other protocols within the existing Internet Protocol (IP) router code.

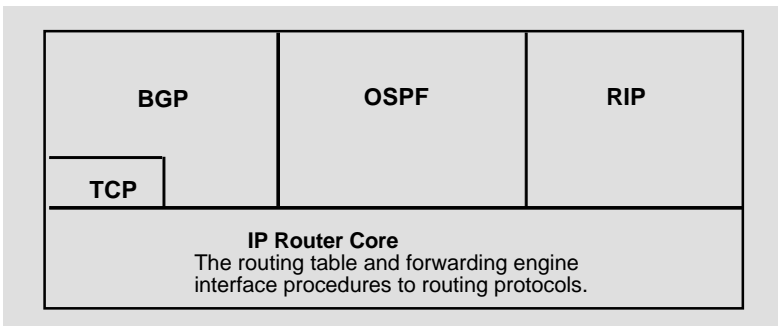


Figure 18. BGP within IP Router Code

BGP systems initially exchange routing information by sending an entire BGP routing table. Thereafter, only updates are exchanged. Acquired routing information is inserted into the routing table by interface procedures that belong to the core of the IP router. Other protocol routes, such as OSPF or RIP, are available for the BGP through the common IP routing table.

A BGP router that sends messages is called a *BGP speaker* or a *peer*. Other active peers that belong to the same autonomous system as the BGP speaker are called *internal peers*. Active peers that belong to autonomous systems other than that of the BGP speaker are called *external peers*.


When a BGP speaker receives a new route advertisement from a peer over a BGP link (that is, from another autonomous system), it advertises this route to all other configured, active speakers (both within the autonomous system and in other autonomous systems), as long as the route is better than other known routes to the advertised network or if no other acceptable routes are known. Unreachable routes are also advertised. This information along with the path attributes is contained in the *update* message. A BGP speaker must generate an update message to all peers when it selects a new route.

BGP uses a variety of message types (indicated by the *type* field in the message header) and they are as follows:

- *open* - establishes a relationship with a peer BGP router.
- *update* - exchanges routing information between BGP peers.
- *keep-alive* - ensures BGP peers are up and running
- *notification* - exchanges diagnostic information between BGP peers.

Specifying the Import and Export Policies

Not all routes learned by a protocol are loaded (imported) into the IP routing table. There are BGP routes that are present in received BGP advertisements but are intentionally left unused. The process of whether or not to load a route into the routing table is called the *Import Policy*. The following Import Policy parameters can be set through ClearSight or by console commands:

 **Note:** The described parameters determine the Import Policy. You can set all of them, or a subset of them, but you must set at least one of the parameters to establish the Import Policy.

- The net range (IP address and mask) to which routes should belong.
- The AS from which the routes are received.
- The IP address of a peer router from which the routes should be received.
- The number of an AS that originated a route.
- The origin of the routes.
- The weighted preference for each route that is imported.

Once you set the parameters, you can decide whether the routes that match the policy are to be imported or ignored.

Routes learned by one protocol and installed in the routing table can be advertised by another protocol. For example, BGP can advertise routes acquired by RIP or OSPF. The following describes in details how this function works:

- First, following checkings are performed:
 1. do split horizon - if the peer from which we learnt the route is external (i.e., in another AS) then do not advertise it back to this peer
 2. if the route is received from internal peer (i.e. in our Autonomous System) then do not advertise to other internal ones. IGP protocols like OSPF/RIP are responsible for distributing intra-AS routes.
 3. if the route is a CIDR one (i.e. the IP mask is shorter than the natural mask) or it is an aggregate then do not advertise it to peers running version 3 of BGP
 4. do not advertise routes which are directed broadcasts for local networks. Do not advertise DISCARD routes (what includes natural networks created by RIP). The shared network with the peer is not advertised too.
- For all other routes, the process of the Export Policy determines whether or not to advertise a route by a protocol. Routes learned by one protocol and installed in the routing table can be advertised by another protocol. For example, BGP can advertise routes acquired by RIP or OSPF. Again, as with the Import Policy, the following Export Policy parameters can be set through ClearSight or by console commands:

► **Note:** The described parameters determine the Export Policy. You can set all of them, or a subset of them, but you must set at least one of the parameters to establish the Export Policy.

- The net range (IP address and mask) to which a route should belong.
- The AS number to which to export a route.
- The IP address of a peer to which to export a route.
- The protocol-specific information that a non-BGP route should have.
- The protocol from which routes when present in the routing table should be learned and advertised. For example, you can make BGP learn and advertise RIP routes but not OSPF routes or learn and advertise OSPF routes but not RIP routes.
- The metric with which a route is exported to BGP.

Once you set the parameters, you can decide whether the routes that match the policy are to be exported or blocked.

Each protocol instance (that is, RIP, OSPF, and BGP) has its own Import and Export policy. For details on setting Import and Export Policy parameters refer to the ClearSight or console commands documentation.

- **Note:** If a route matches for more than one Export/Import Policy definition, the policy with the highest precedence is selected. A policy has a higher precedence than another if the sum of its priorities is greater.

Path Selection

IP router's forwarding engine picks the best route to a particular destination from routes installed in the routing table. With IGP protocols routes are qualified according to their preference and metric attributes. Across the AS boundary the qualifying rules differ, metrics usually are not comparable, and the preferences of the routes that are installed in the routing table are calculated according to the import policy.

The best path is picked using (in order of significance):

1. **Preference**
A lower value is better. The range of values is 0..255, with 255 indicating a route not used.
2. **Metric**
Applies to routes coming from the same AS. A lower metric is better.
3. **Origin**
From best to worst, the origin values are: IGP, EGP, and INC.
4. **AS path weight**
Equivalent to the AS path length. A lower path weight is better. Path weighting affects the best route choice when all other factors are equal. The range of values is 0..255, with 128 being the default.
5. **Router ID**
A lower router ID is better.

BGP Terminology and Concepts

This section provides definitions for terms and concepts that are specific to the BGP.

Autonomous System (AS)

A set of routers that use one or more interior gateways protocols managed under a single administration. An AS appears to other ASs to have a single coherent interior routing plan and presents a consistent picture of the destinations that are reachable through it.

Classless Inter-Domain Routing

A mechanism that slows the growth of the IP routing tables and thus reduces the need for allocating new IP network numbers.

IP Router Core

A section of Olicom's IP router software that consists of the forwarding engine, interface procedures to the routing protocols, and procedures that maintain the IP routing table.

Exterior Gateway Protocol (EGP)

A class of protocols that provides routing information between autonomous systems. BGP is an example of such protocols.

Interior Gateway Protocol (IGP)

A class of protocols that provide routing information within an autonomous system. OSPF and RIP are examples of such protocols.

Management Information Base (MIB)

A database of information on managed objects that can be accessed by the SNMP network management suite.

Open Shortest Path First (OSPF)

A link-state, hierarchical IGP routing protocol.

Peer

A BGP router (also refer to as a *BGP speaker* or a *neighbor*). BGP supports two different kinds of peers - internal and external. Internal peers are in the same autonomous system. External peers are in other autonomous systems.

Protocol Instance

A part of Olicom's IP router that is responsible for originating and processing routing protocol packets. Information gathered by such an instance updates the IP routing table. There can be several protocol instances within the IP router (for example, RIP, OSPF, and BGP instances).

Route Aggregation

A process that combines the attributes of several different routes so that a single route can be advertised. Route aggregation reduces the amount of information that a router stores and exchanges with other BGP routers.

Routing Information Protocol (RIP)

A distance-vector IGP routing protocol.

Simple Network Management Protocol (SNMP)

A standard protocol suite used for network management.

Transmission Control Protocol (TCP)

A connection-oriented, reliable data transfer protocol that runs over IP. BGP uses TCP as its transport layer.



6. TCP

This chapter discusses the Transmission Control Protocol (TCP) as implemented by Olicom.

In Olicom routers, TELNET, BGP, and DLSw all use TCP as their transport protocol. They are described in separate chapters of this manual.

Sections

- *Overview*
- *Technical Discussion*

Overview

The Transmission Control Protocol is the Internet standard for reliable data transfer over a network. The most significant features of TCP are as follows:

- TCP is a connection-oriented protocol, in that communication between two endpoints proceeds through three phases: connection establishment, data transfer, and connection release.
- TCP is a reliable transfer protocol, in that TCP recovers from data that is damaged, lost, duplicated, or delivered out of order by the underlying network system. This is accomplished by assigning a sequence number to each transmitted byte, and requiring a positive acknowledgment (ACK) from the receiving TCP. If the ACK is not received within a time-out interval, the data is retransmitted. At the receiver, the sequence numbers are used to correctly order segments that may be received out of order and to eliminate duplicates. Damage is handled by adding a checksum to each segment transmitted, checking it at the receiver, and discarding damaged segments.
- TCP accepts data from applications in a stream-oriented fashion. TCP guarantees that all the data will be delivered to the other end in the same order it was sent, and without duplicates. This is in contrast to lower level protocols which are designed to send frames or datagrams.
- TCP supports flow control, which makes it possible to prevent buffer overrun and possible congestion on the receiving machine. The flow control mechanism works by returning a “window size” with every ACK. This size indicates how many bytes are acceptable beyond the last segment successfully received (i.e., how many bytes a sender may transmit before receiving further permission).

- TCP supports a facility for multiplexing multiple user sessions within a single machine. To that end, TCP provides a set of ports within each host. Concatenated with the IP addresses of the machine, it forms a socket. Because a pair of sockets uniquely identifies a connection, the same server socket can be simultaneously used in multiple sessions with different clients.

TCP is designed to run over IP. User application programs such as TELNET and FTP use TCP as their reliable transport layer protocol.

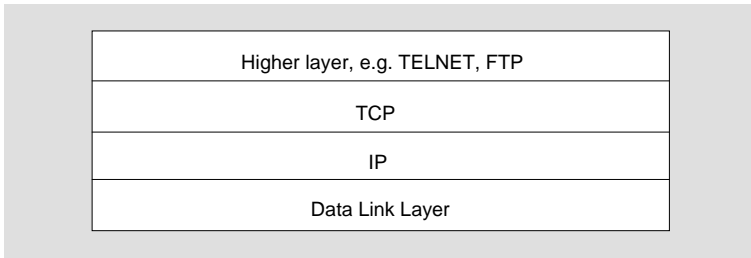


Figure 19. Protocol Layering

As implemented by Olicom, TCP is compliant with the Internet standard described in RFC793.

Technical Discussion

This section discusses the following topics:

- TCP Connections and Ports
- Sliding Window
- Time-outs, Retransmissions
- TCP Segment Format
- Passive and Active Opens
- Closing a TCP Connection
- MSS Option
- TCP Checksum
- TCP State Machine
- Glossary of Terms

TCP Connections and Ports

As mentioned before, a TCP connection is identified by a pair of sockets. Each socket consists of the machine's IP address and a TCP port number. In the client-server architecture used by such applications as TELNET or FTP, a server usually listens for incoming connection requests on a well-known port. This simplifies session establishment. On the other connection side, a client can independently select a TCP port number for that connection.

Table 7 lists some of the assigned TCP port numbers.

Port Number	Description
20	FTP-DATA, File Transfer Protocol (data)
21	FTP
23	TELNET, Terminal Connection
25	SMTP, Simple Mail Transport Protocol
53	Domain Name Server
111	SUN Remote Procedure Call
179	BGP, Border Gateway Protocol

Table 7. TCP port numbers

More information on assigned port numbers can be found in “Assigned Numbers” - RFC1700.

Sliding Window

For flow control, TCP uses a sliding window mechanism which makes it possible to send multiple segments before an acknowledgment arrives. Such an approach increases the throughput by keeping the network busy.

The TCP sliding window operates at the byte level, not at a packet level. In other words, the window size is expressed in a number of bytes. Each endpoint has two windows: the **receive window**, which represents the sequence number of bytes the local TCP is willing to receive, and the **send window**, which represents the sequence number of bytes which the remote TCP is willing to receive.

Figure 20 shows an example slide window for the data to be sent. The other endpoint of the connection has a similar window for received data.

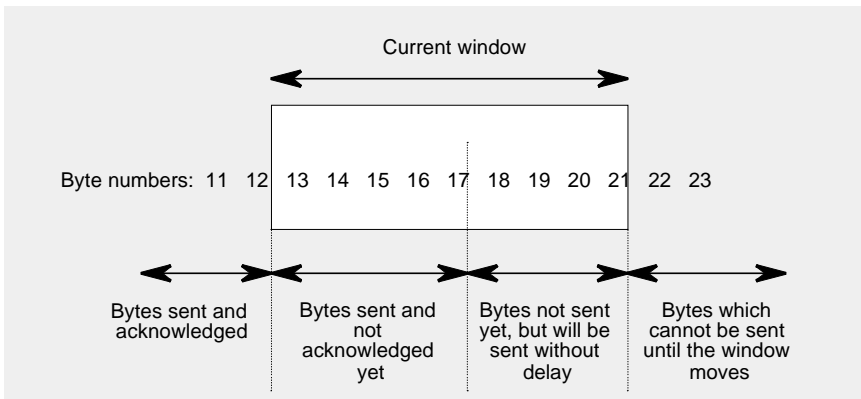


Figure 20. TCP sliding window

Because TCP is a full-duplex transport protocol, each end-point has its own send and receive window.

To improve flow control, TCP allows the window size to vary. Each acknowledgment (ACK), which specifies how many bytes the other side already received, contains also a window advertisement. This window advertisement specifies how many additional bytes the other machine is ready to accept. If the advertised window increases, the sender can increase its send window to transmit more bytes without an ACK. If the advertised window decreases, the sender should decrease its send window too, and stop sending data if the window is exceeded.

Time-outs and Retransmissions

When a sender does not receive an acknowledgment (ACK) for the segments it sent, and a time-out occurs, then the unacknowledged data is retransmitted. TCP does not use a fixed retransmission timer. Rather, the time-out value is derived from an analysis of the delay in receiving an acknowledgment from the other side of the connection.

TCP Segment Format

A stream of bytes an application such as TELNET or FTP tries to send over a TCP connection is divided into segments for transmission over a network. Such segments are encapsulated into IP packets and in this form sent to the other side of a connection. A TCP header follows the IP header, supplying information specific to the TCP protocol. The TCP header format is described in Figure 21.

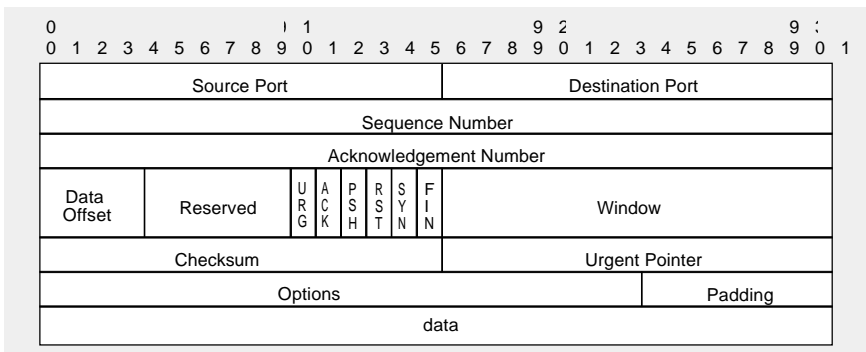


Figure 21. TCP Header Format

Header Segment	Description
Source Port: 16 bits	The source port number.
Destination Port: 16 bits	The destination port number.
Sequence Number: 32 bits	The sequence number of the first data octet in this segment (except when SYN is present). If SYN is present, the sequence number is the initial sequence number (ISN) and the first data octet is ISN+1.

Table 8. TCP Header Details

Acknowledgement Number: 32 bits	If the ACK control bit is set, this field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established, this is always sent.
Data Offset: 4 bits	The number of 32 bit words in the TCP Header. This indicates where the data begins. The TCP header (even the one including options) is an integral number of 32 bits long.
Reserved: 6 bits	Reserved for future use. Must be zero.
Control Bits: 6 bits (from left to right)	URG: Urgent Pointer field significant ACK: Acknowledgment field significant PSH: Push Function RST: Reset the connection SYN: Synchronize sequence numbers FIN: No more data from sender
Window: 16 bits	The number of data octets beginning with the one indicated in the acknowledgment field which the sender of this segment is willing to accept.
Checksum: 16 bits	The TCP header checksum.
Urgent Pointer: 16 bits	The current value of the urgent pointer as a positive offset from the sequence number in this segment. The urgent pointer points to the sequence number of the octet following the urgent data. This field is only interpreted in segments with the URG control bit set.
Options: variable	Options may occupy space at the end of the TCP header and are a multiple of 8 bits in length. All options are included in the checksum. An option may begin on any octet boundary. The currently defined options include: <End of option list>, <No-Operation> and <Maximum Segment Size>.
Padding: variable	The TCP header padding is used to ensure that the TCP header ends and the data begins on a 32-bit boundary. The padding is composed of zeros.

Table 8. TCP Header Details

Passive and Active Opens

Because TCP is a connection-oriented protocol, both sides of a connection must agree to participate in it. Usually it is accomplished in the following way: One of the endpoints makes a passive open (i.e., tells its operating system that it wants to accept incoming connection requests.) The other side performs an active open - it simply sends a connection request. When the passive endpoint receives that request, it may accept it or reject. If the request is accepted, the connection is established and the application may exchange data.

The procedure of establishing a connection is shown in Figure 22.

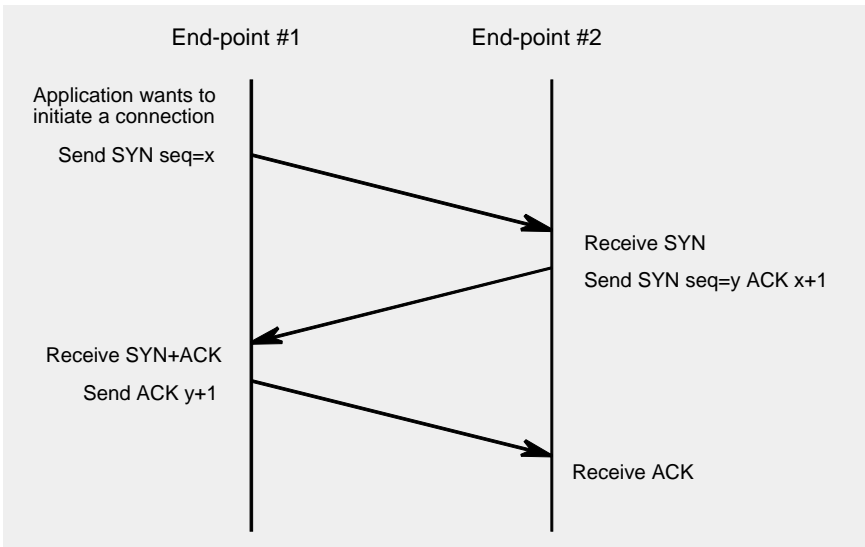


Figure 22. Establishing a TCP connection

The initial sequence numbers are carried by SYN segments. These numbers are also acknowledged during session establishment, which ensures consistency between both sides of the connection. The initial sequence numbers are chosen randomly, because they can not always start at the same value.

Closing a TCP Connection

If one side of a connection has no more data to send, it can close the connection. Because TCP is full-duplex, the machine which closes a TCP connection may continue to receive data until it is told that the other side has also closed the connection. Once a connection has been closed in a given direction, TCP refuses to accept any data from that direction.

Figure 23 shows the procedure for closing a TCP connection.

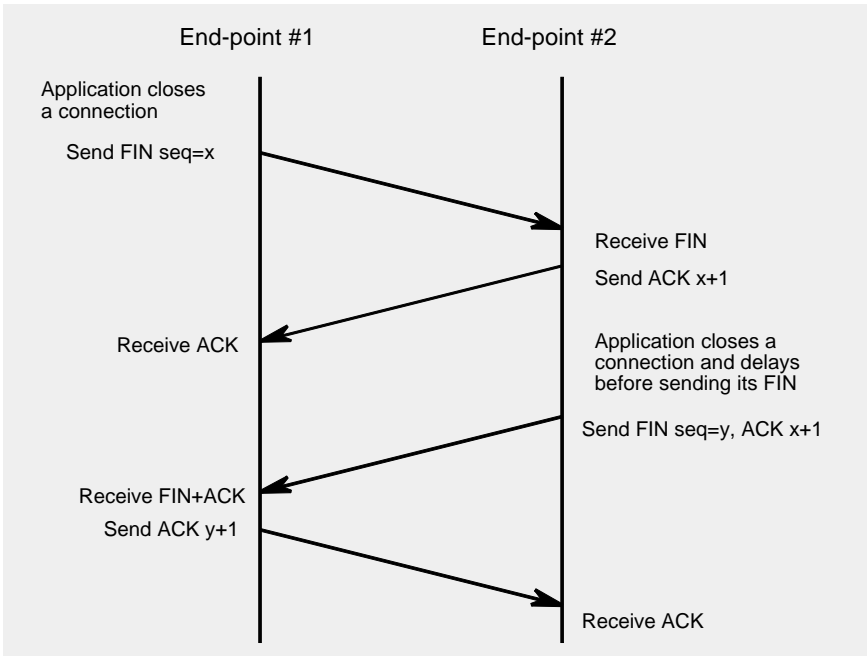


Figure 23. Closing a TCP connection

Normally, a TCP application uses the described mechanism to close a connection. Sometimes however, abnormal conditions force an immediate break in the connection. If one end point sends a segment with the RST bit set, the other side aborts the connection immediately.

MSS Option

Both sides of a TCP connection must agree on the maximum segment size (MSS) they can transfer. TCP uses the MSS option to negotiate this value. The negotiation takes place only at the initial phase - connection establishment. Usually, the MSS value is determined such that IP packets generated by TCP end-points do not require IP fragmentation (i.e., the MSS matches the minimum MTU along the path between two communicating machines).

Each side can send segments smaller than the negotiated MSS.

TCP Checksum

The checksum field in the TCP header is used to verify the integrity of the received segments.

The checksum value is computed as the 16-bit one's complement of the one's complement sum of all 16-bit words in the segment. If a segment contains an odd number of bytes, then one byte with zeros is padded to form a 16 bit word for checksum purposes. The pad is not transmitted as part of the segment. While computing the checksum, the checksum field itself is replaced with zeros. The checksum also covers a 96 bit pseudo header conceptually prefixed to the TCP header. This pseudo header contains the Source IP Address, the Destination IP Address, the Protocol, and TCP length. This allows to check the integrity of misrouted segments.

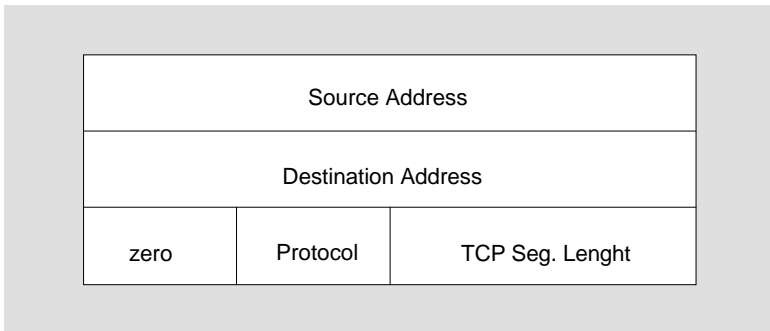


Figure 24. TCP pseudo-header

TCP Finite State Machine

Each TCP connection is described by a finite state machine (FSM). During its lifetime, a connection changes its state according to events, including user calls such as OPEN, SEND, RECEIVE, CLOSE; incoming segments, particularly those containing the SYN, ACK, RST and FIN flags; and time-outs.

Table 9 briefly describes the states of the TCP FSM. The TCP FSM is shown in Figure 25.

State	Description
LISTEN	TCP is waiting for a connection request from any remote TCP.
SYN-SENT	TCP is waiting for a matching connection request after a connection request has been sent.
SYN-RECEIVED	TCP is waiting for a confirming connection request acknowledgment after having both received and sent a connection request.
ESTABLISHED	This state describes an open connection. Data may be received and send to/from an upper application.
FIN-WAIT-1	TCP is waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent.
FIN-WAIT-2	TCP is waiting for a connection termination request from the remote TCP.
CLOSE-WAIT	TCP is waiting for a connection termination request from the local user.
CLOSING	TCP is waiting for a connection termination request acknowledgment from the remote TCP.
LAST-ACK	TCP is waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request).
TIME-WAIT	TCP is waiting for enough time to pass to be sure that the remote TCP received the acknowledgment of its connection termination request.
CLOSED	represents no connection state at all.

Table 9. States of the TCP FSM

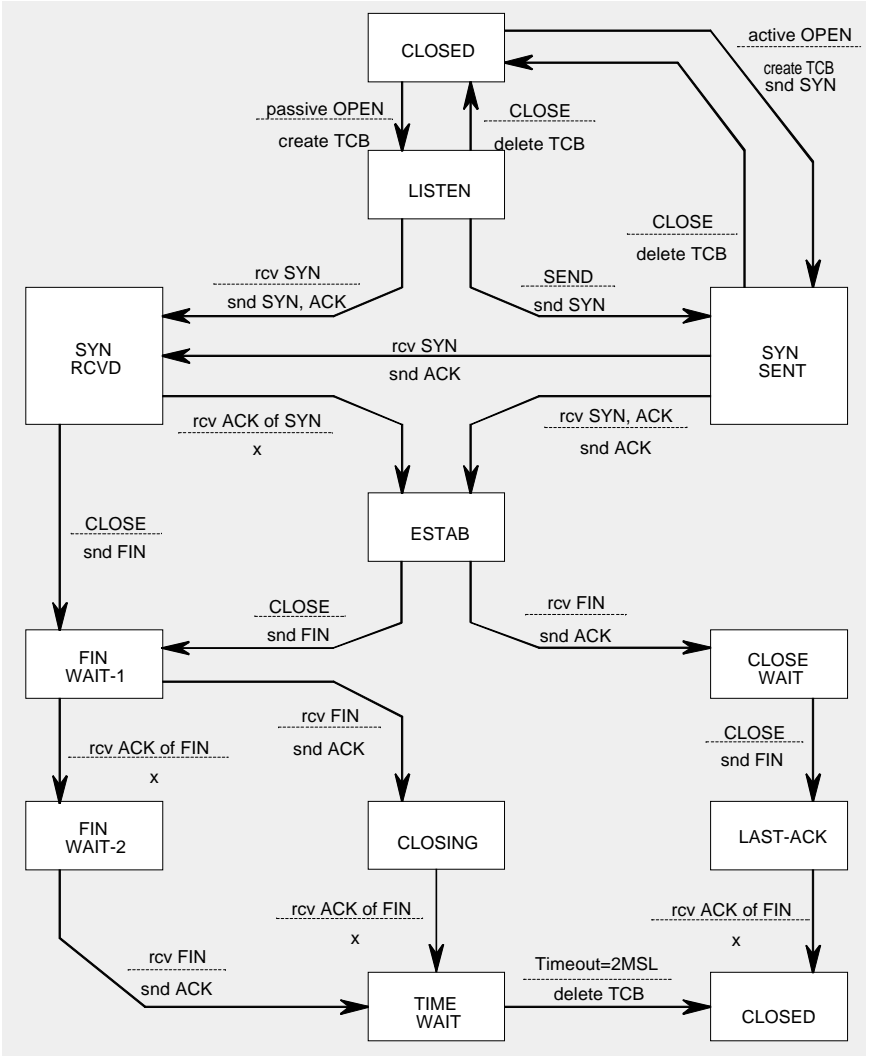


Figure 25. TCP Finite State Machine

TCP Glossary

ACK

A control bit (acknowledge) occupying no sequence space, which indicates that the acknowledgment field of this segment specifies the next sequence number the sender of this segment is expecting to receive, hence acknowledging receipt of all previous sequence numbers.

connection

A logical communication path identified by a pair of sockets.

datagram

A message sent in a packet switched computer communications network.

FIN

A control bit occupying one sequence number, which indicates that the sender will send no more data or control occupying sequence space.

fragment

A portion of a logical unit of data, in particular an internet fragment is a portion of an internet datagram.

header

Control information at the beginning of a message, segment, fragment, packet or block of data.

ISN

The Initial Sequence Number. The first sequence number used on a connection. Usually selected on a clock based procedure.

Options

An Option field may contain several options, and each option may be several octets in length. The options are used primarily in testing situations; for example, to carry timestamps. Both the Internet Protocol and TCP provide for options fields.

port

The portion of a socket that specifies which logical input or output channel of a process is associated with the data.

PUSH

A control bit occupying no sequence space, indicating that this segment contains data that must be pushed through to the receiving user.

receive next sequence number

This is the next sequence number the local TCP is expecting to receive.

receive window

This represents the sequence numbers the local (receiving) TCP is willing to receive. Segments containing sequence numbers entirely outside of this range are considered duplicates and discarded.

RST

A control bit (reset), occupying no sequence space, indicating that the receiver should delete the connection without further interaction.

segment

A logical unit of data, in particular a TCP segment is the unit of data transferred between a pair of TCP modules.

segment acknowledgment

The sequence number in the acknowledgment field of the arriving segment.

segment length

The amount of sequence number space occupied by a segment, including any controls which occupy sequence space.

segment sequence

The number in the sequence field of the arriving segment.

send sequence

This is the next sequence number the local (sending) TCP will use on the connection. It is incremented for each octet of data or sequenced control transmitted.

send window

This represents the sequence numbers which the remote (receiving) TCP is willing to receive. It is the value of the window field specified in segments from the remote (data receiving) TCP.

socket

An address which specifically includes a port identifier, that is, the concatenation of an Internet Address with a TCP port.

SYN

A control bit in the incoming segment, occupying one sequence number, used at the initiation of a connection, to indicate where the sequence numbering will start.

TCP

Transmission Control Protocol: A host-to-host protocol for reliable communication in internetwork environments.

URG

A control bit (urgent), occupying no sequence space, used to indicate that the receiving user should be notified to do urgent processing as long as there is data to be consumed with sequence numbers less than the value indicated in the urgent pointer.

urgent pointer

A control field meaningful only when the URG bit is on. This field communicates the value of the urgent pointer which indicates the data octet associated with the sending user's urgent call.



7. IPX Routing

This chapter explains Olicom's Internetwork Packet Exchange (IPX) Routing.

Sections

- *Overview*
- *Technical Discussion*
- *Network Considerations*

Overview

Olicom's IPX Routing supports the Novell IPX Protocol and provides all the features of the Novell IPX router. Olicom's IPX router directly connects Token Ring and Ethernet ports, through wide area network (WAN) adapters (T1 and serial interfaces) or X.25 adapters. Each Ethernet port, Token Ring port, Frame Relay port, WAN port, or X.25 port (virtual circuit) is treated as a separate interface. When IPX Routing is enabled, the Olicom router operates as a bridge-router. IPX packets are routed while other packets are bridged. The IPX Routing feature Version 1.0 is provided by Olicom as a Feature Pack upgrade.

An Olicom router with the IPX Routing capability offers the following features:

- Starting with XL version 4.1, Olicom IPX Router is fully compatible with Novell's IPX Router Specification version 1.20 and Novell's IPX Router Tests version 1.20.
- From XL version 4.1, IPX Routing allows the creation of multiple IPX circuits on Ethernet or Token Ring interfaces, and of a single IPX circuit on other interfaces. Each IPX circuit on an interface is associated with a specific IPX frame type.
- Allows Simple Network Management Protocol (SNMP) management of all IPX Router parameters. In addition, Novell's IPX and Routing Information Protocol/Service Advertising Protocol (RIP/SAP) management information bases (MIBs), as well as an Olicom private MIB, are supported.
- Allows enabling and disabling of IPX Routing functions globally or on specific ports (IPX circuits).
- Allows split traffic on equivalent routes. (You can configure the maximum number of split paths.)

- Allows enabling and disabling of RIP/SAP packet processing globally and on specific ports (IPX circuits).
- Allows you to set a maximum number of registered network and server paths. This saves memory used to store additional paths to improve RIP/SAP protocol performance.
- Allows Reduced (*Smart*) Advertising on Slow Interfaces, which is the option of sending advertising packets only when registered IPX network and server information has changed. This addresses performance degradation that can occur on slow interfaces when sending advertising packets too frequently. This feature can be configured on selected ports (IPX circuits). From XL version 3.0, RIP Reduced Advertising and SAP Reduced Advertising are configured separately. From XL version 4.1, an enhanced form of reduced advertising (Acknowledged Reduced Advertising) is available. It guarantees, through acknowledgments, that the IPX router on the other side of a point-to-point line will be notified of all changes in the network/server table.
- Allows processing of IPX Source Routed packets on selected ports (IPX circuits). This provides the following:
 - Receipt of IPX Single Route Broadcast (SRB) packets
 - Transmission of RIP and SAP packets advertised as SRB packets
 - Transmission of non-RIP and non-SAP packets that advertise with or without SRB
- Allows automatic setting of IPX network numbers. IPX Router automatically learns the proper IPX network number for all IPX router ports (IPX circuits) connected to networks with established IPX network numbers.
- Allows configuration of the IPX frame format for the following interfaces:
 - For Ethernet -- 802.3 Ethernet (Novell Ethernet) standard frame format, Ethernet II (Portable Netware), Ethernet 802.2 or Ethernet SNAP standard frame format
 - For Token Ring -- Token Ring 802.2 and Token Ring SNAP
- For other interfaces -- In XL version 4.1 and later, 802.3 Ethernet, Ethernet II, Ethernet 802.2, Ethernet SNAP, Token Ring 802.2, and Token Ring SNAP standard frame format.
- Starting with XL version 4.1, on Ethernet and Token Ring interfaces it is

possible to associate multiple IPX circuits to a single interface and to assign a different frame format to each of the circuits (other interface types can have just a single IPX circuit and frame format).

- Allows positive and negative filtering of specific networks and servers that receive or advertise through selected ports (IPX circuits).
- Allows setting of static networks and static servers to force selected paths. By disabling RIP/SAP packet processing on selected ports (IPX circuits) and setting relevant static networks and static servers, you eliminate RIP/SAP protocol traffic on slow lines. Static routes can be registered through different ports (IPX circuits), making it possible for IPX traffic to quickly switch (for example, using, Dial Backup) to an alternate path if the current port (IPX circuit) fails.
- Allows path metrics (delays and hops) to change for registered networks and servers. This forces selected paths registered through specified ports (IPX circuits).
- Allows tuning parameters for RIP and SAP advertisements to different kinds of lines connected to specified ports (IPX circuits).
- Allows limiting of IPX packet size on selected ports (IPX circuits).
- Allows enabling or disabling of SAP Get Nearest Server Reply on specified ports (IPX circuits). This allows you to force a connection to a preferred server via selected routers.
- Allows IPX NetBIOS (IPX Type 20 propagation packets) processing over IPX traffic on specified ports. From XL version 4.1, it is possible to reduce the propagation of IPX NetBIOS packets by not propagating such packets coming from the same source but over different IPX circuits, so that each IPX Type 20 is advertised through the same IPX Router just once.
- Allows monitoring of the IPX router through IPX Global Statistics and IPX Port (Circuit) Statistics.
- Allows monitoring of network conflict errors by notifying you through the ClearSight Event Manager of improperly set IPX network numbers and device addresses.
- Allows unnumbered IPX RIP point-to-point lines. By allowing point-to-point lines to remain unnumbered, it is unnecessary to allocate and configure a pool of IPX numbers for IPX point-to-point links, and the internetwork is not burdened with unneeded RIP information. Unnumbered IPX RIP point-to-point lines are also necessary for, among other things, transmission groups.
- From XL version 4.1, IPX router can spoof IPX watchdog packets and thereby reduce IPX control traffic on a given IPX circuit. Each IPX server periodically

sends watchdog request packets to all IPX workstations logged in and each of those stations answers the server with a watchdog response packet. With spoofing, which is configurable per IPX circuit, the IPX router receives and responds directly to watchdog request packets instead of forwarding requests from the server to the workstations and forwarding responses from the workstations to the server. Use of this option is highly recommended for Dial on Demand lines.

- From XL version 4.1, the reachability of an IPX destination can be checked with an IPX Ping (through ExpertTest) using the IPX Diagnostic protocol or Novell's IPX Ping protocol. Most IPX stations, routers, and servers can respond to the IPX Diagnostic packets, while use of Novell's new IPX Ping protocol is still limited (to servers using IPXPING.NLM, XL 4.1 and later running IPX Router, etc.).
- From XL version 5.1, IPX router can spoof SPX keep-alive packets. SPX Spoofing is especially necessary for Dial On Demand lines when SPX sessions are established through these lines. It allows to reduce SPX traffic forwarded through the point-to-point line and as a result, to disconnect DOD line and save connection cost without breaking running SPX sessions.

Technical Discussion

The NetWare Internetwork Packet Exchange (IPX) Protocol is a variation on the Xerox Network Systems (XNS) Internet Transport Protocol. Performing tasks at the Network Layer of the OSI model, this protocol routes internet packets through the internetwork using store-and-forward systems called internetwork routers. The internetwork router treats each of these packets independently and delivers them by way of the most efficient path. However, there is no guarantee that they are delivered once or only once, nor that they are delivered in the same order they were transmitted. The upper layers of the Novell protocol (for example, SPX) deal with these issues.

An internetwork is a communication system configured to carry internet packets between workstations and servers through internetwork routers. Servers and workstations on the same network communicate without the aid of an internetwork router. An IPX network is assigned an IPX network number which must be unique throughout the entire integrated network.

The standard part of an IPX internet packet consists of two fields, a control field (30 byte header) and a data field, that can be from 0 to 4434 bytes. The IPX packet structure is shown below.

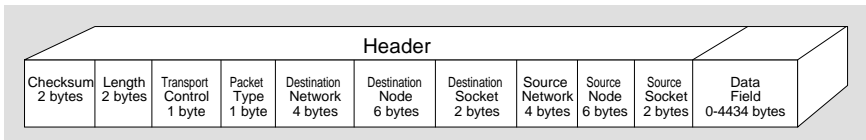


Figure 26. IPX Packet

➔ **Note:** Although NetBIOS is not directly supported by Novell, it provides a NetBIOS emulation using IPX that supplies NetBIOS API while communicating with standard IPX packets. Therefore, NetBIOS packets are sent as a broadcast propagated packet (packet type is 20 dec or 14 hex). The IPX router propagates NetBIOS packets through all local networks (local means a network connected to the relevant interface) through which this packet has not been previously transmitted.

Field	Description
Checksum	The originator of an IPX packet can include a checksum. A checksum of FFFF hex indicates that the originator chose not to compute the checksum. Otherwise, the checksum is calculated with the values of all bytes in the frame (starting with the Checksum field itself and ending with the last byte of the IPX data), except the Checksum and Transport Control fields which are treated as containing zeros.

Table 10. IPX Packet Header

Length	Contains IPX packet length, which is the length of the header plus the length of the data (minimum length = 30 bytes, and maximum length = 4464 bytes). For LANs it is higher and depends on the transmission media.
Transport Control	Contains the number of intermediate networks through which the IPX packet was routed.
Packet Type	Indicates the type of service offered or required by the packet. The following values have been defined by Xerox: 0 Unknown Packet Type 1 Routing Information Packet (RIP) 2 Echo Packet 3 Error Packet 4 Packet Exchange Packet (PXP) or Service Advertising Packet (SAP) 5 Sequenced Packet Protocol Packet (SPX) 17 Netware Core Protocol Packet (NCP) 20 Propagated Packet (IPX Packet Type 20 such as NetBIOS over IPX Broadcast Packet)
Destination Network	Contains the IPX network number of the network to which the destination device belongs. When this field is 0, the destination device is on the same network as the source device and the packet is not routed.
Destination Node	Contains the physical address of the destination device. (Address FF-FF-FF-FF-FF-FF hex is broadcast to all devices on the destination network.)
Destination Socket	Contains the socket address of the destination. Sockets route packets to different processes within a single device. The NetWare hexadecimal socket numbers are: 451 Netware Core Protocol process 452 Service Advertising Protocol process 453 Routing Information Protocol process 455 Novell NetBIOS process 456 Diagnostic process 4000÷7FFF Dynamic sockets used by workstations for interaction with file servers and other network communications. 8000÷FFFF Sockets assigned for use by registered software developers writing application packages for NetWare. 9001 Netware Link Services Protocol process 9004 IPX WAN Version 2 Protocol process 9086 IPX Ping Protocol process

Table 10. IPX Packet Header

Source Network	Contains the network where the source device belongs. If set to zero, the physical network of the source is unknown.
Source Node	Contains the physical address of the source device.
Source Socket	Contains the socket address of the process that transmits the packet. Source socket numbers follow the same conventions as those for destination sockets.
Data	Contains 0 to 4434 bytes of packet data.

Table 10. IPX Packet Header

Best Route Determination

IPX packets are routed through the internetwork based on the most efficient path. Two concepts are important to understanding how the best route (path) is determined, *hops* and *delay*.

Hops are the number of intermediate IPX networks (number of other IPX routers) through which the packet must pass to reach the destination IPX network. Internet packets may be passed through a maximum of 16 routers. When the number of hops is equal to 16 (in the RIP/SAP network/server entry or in the network/server table), it is an indication that the specified IPX network or IPX server is in a dead state. This condition appears when a route to the network or server has been timed-out or when a server or router has been downed.

Delay is the estimated time, in 50-ms ticks, necessary to deliver a 576 byte packet from the IPX router to a node on the destination network. Delay is dependent on the type of transmission media.

The best route (path) to the specified IPX network is the route registered with the least delay and then the least number of hops. If more than one route with equal delay and equal hops has been registered through different interfaces, then the last registered route is treated as the best route. Consider the following examples:

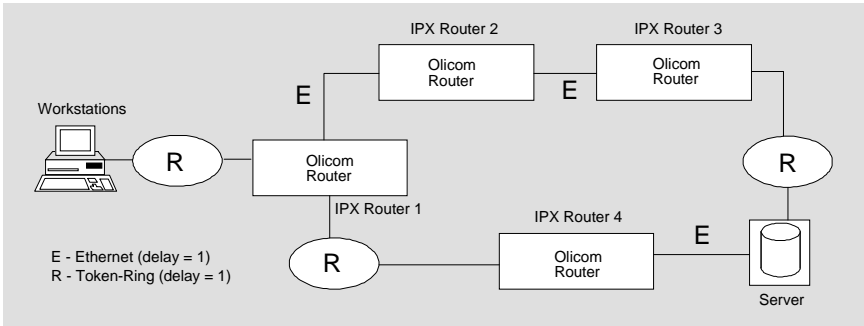


Figure 27. Best Route Determination -- Example 1

Route from Workstation to Server	Delay	Hops	Best Route
Through IPX Routers 1, 2 and 3	4	3	
Through IPX Routers 1 and 4	3	2	✓

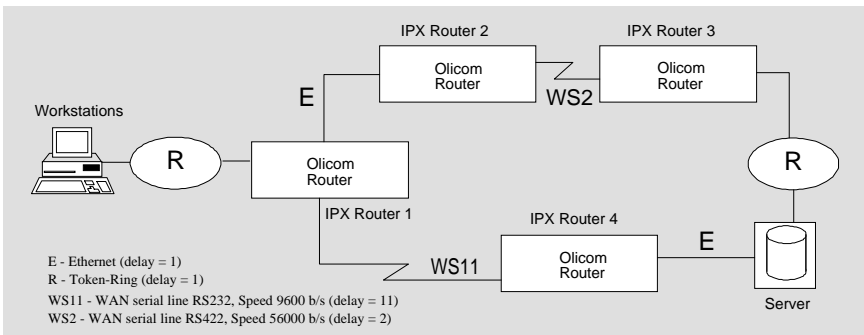


Figure 28. Best Route Determination -- Example 2

Route from Workstation to Server	Delay	Hops	Best Route
Through IPX Routers 1, 2, and 3	5	3	✓
Through IPX Routers 1 and 4	13	2	

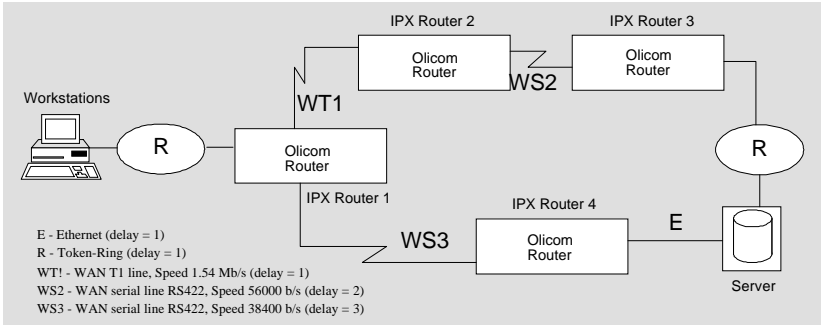


Figure 29. Best Route Determination -- Example 3

Route from Workstation to Server	Delay	Hops	Best Route
Through IPX Routers 1, 2, and 3	5	3	
Through IPX Routers 1 and 4	5	2	✓

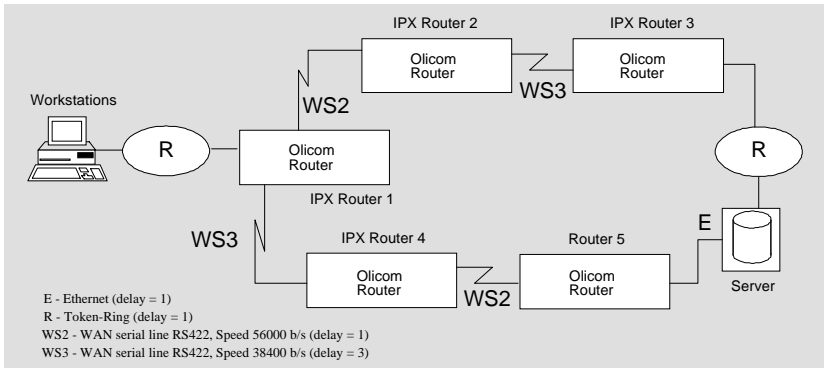


Figure 30. Best Route Determination -- Example 4

Route from Workstation to Server	Delay	Hops	Best Route
Through IPX Routers 1, 2, and 3	7	3	*
Through IPX Routers 1, 4, and 5	7	3	

*Both routes (paths) are equal in terms of hops and delays. Most likely, the route through IPX Routers 2 and 3 are registered last because it takes Token Ring longer

than Ethernet to become active. To force a route through IPX routers 1, 4, and 5 as the best route, set the relevant Delay Offset In and Delay Offset Out (or Hop Offset In and Hop Offset Out) on the port connected to WS2 Line (on Router 1 or Router 2) or to WS3 line (on Router 2 or Router 3).

The above examples show that the IPX router must maintain information about servers and the networks on which these servers reside. The IPX router maintains two tables for this purpose: the routing (network) table and the server table. The routing table is maintained through the Routing Information Protocol and the server table is maintained through the Service Advertising Protocol.

The server table allows clients on the network to determine which services are available on the network and to obtain the IPX address (network number, node number, and socket) of the servers where they can access those services. Session packets are then addressed to the destination IPX address, which means that the routing decision is based on the routing (network) table.

Routing Information Protocol

IPX routers use the Routing Information Protocol to inform one another of the internetwork topology. This information is communicated via Routing Information Protocol (RIP) packets which are used to maintain the routing table. RIP packets are identified by socket 0x453 which Novell has designated for all RIP operations.

RIP packets can be sent to a specified device or as a broadcast packet. There are two types of RIP packets, RIP Request packets and RIP Response packets.

RIP Request packets may contain a request to Return All Known Networks or to Return Specified Network Information. A Return All Known Networks request (All Route Request) is sent through the specified port (IPX circuit) when this port becomes active or through all active ports (IPX circuits) when a Reset router command is entered. Return Specified Network Information request is sent from the workstation when this workstation attaches to the specified server.

A RIP Response packet contains either single or multiple Network Information Entries. Each entry consist of IPX network number and the number of hops and delay (number of 50-ms. ticks) of the best route registered for the specified network (see also *Best Route Determination* on page 83). A RIP Response may be sent out as a unicast packet or as a broadcast packet. A unicast response packet is generally used when the router is responding to a previous request. A broadcast response is generally an advertisement packet.

The IPX router advertises all known network information by broadcasting Routing Information packets every RIP update interval. These packets have the same format as RIP Response packets and are sent through all active ports (IPX circuits). From these packets, the IPX router learns new routes (paths) and updates previously registered routes. When a new route is registered, the IPX router

recalculates the best route for a given network. If there is a new best route, the IPX router advertises this new route on the integrated network.

For all RIP response packets, a Split Horizon algorithm is applied: IPX Router does not advertise a network number through the port (IPX circuit) from which the best path to that network (or an equivalent path with the same delay metric) is registered.

If a route has not been updated within (RIP age multiplier * RIP update interval), then a timed-out state is set for this route. If a Network Information Entry is received with the number of hops equal to 16 or a route has been timed-out, then this route is removed from the routing table, the best route for the specified network is updated, and the new state is advertised. If all routes for a given network have been removed, then a dead state (hops = 16) is advertised via a Routing Shutdown Packet, and the specified network structure is removed from the routing table. A Routing Shutdown Packet is also sent when a Down router command is entered. (A routing shutdown packet is a set of the Network Information Entries, with hops = 16, within a normal RIP Response packet.)

Service Advertising Protocol (SAP)

The Service Advertising Protocol (SAP) is used by IPX routers, servers and processes to inform clients of a servers presence on the integrated network. The Service Advertising Protocol makes it possible for clients to easily identify the name and type of servers present of the network. SAP packets are identified by Socket 0x452 which Novell has dedicated for all SAP operations.

There are two types of SAP packets: SAP Response packets and SAP Query packets. A SAP Response packet can be a Server Identification Packet, a General Service Response Packet, a Nearest Service Response, a Servers Advertising Packet, or a Server Shutdown Packet. A SAP Query Packets can be a General Service Query Packet or a Nearest Service Query Packet.

After a server opens the SAP socket, it broadcasts an identification packet on the socket every SAP Update Interval. These Server Identification Packets are received by all IPX routers and servers on the integrated network. Each packet contains single or multiple Server Information Entries. Each entry contains the following information:

Field	Description
Server Type	This field identifies the type of service that the server provides. (Server types can be obtained from Novell.)
Server Name	This is a unique object name that is assigned to the server (48 bytes).
Network	This is the IPX network number on which the server resides.

Node	This field contains the node address of the server.
Socket	This is the socket number on which the server receives service requests.
Hops	This field contains the number of intermediate IPX networks that the Server Identification Packet must traverse when traveling from the server to the client.

From these packets, the IPX router learns about new servers and updates registered servers in the server table. When information about a new server is received by the IPX router, the registered network number is looked up in the routing table. If this network is registered through the same port (IPX circuit) and the hops are equivalent, the server is registered in the server table. If a better or equal path (route) to the registered server is received by the IPX router, the existing path is replaced by the new path and the change is advertised. Every SAP update interval, the IPX router sends a Server Advertising Packet in the form of multiple Server Information Entries.

If the path to the registered server has not been updated within (SAP Age Multiplier * SAP Update Interval), a timed-out state is set for this path. If a Server Information Entry is received with

hops = 16, or if the specified path is timed-out, this path (route) is removed from the server table, the best path for a specified server is updated, and the new state is advertised. If all paths for a given server have been removed, a dead state (hops = 16) is advertised through a Server Shutdown Packet and the server is removed from the server table. A Server Shutdown Packet also is sent when the Down or Reset router commands are entered. (A shutdown packet is a set of Server Information Entries, with hops=16, within a normal Server Advertising Packet.)

IPX routers and Client workstations can use a General Service Query to build a list of all available servers on the integrated network. The query can specify all servers of a particular type or all servers of any type. A General Service Query causes a General Service Response (in the same format as a Server Identification Packet) from every qualified server and IPX router.

Client workstations can also broadcast a Nearest Service Query packet to find the nearest server of a particular type. When an IPX router receives a Nearest Service Query Packet, it responds with a Nearest Service Response (Reply) packet (in the same format as a Server Identification Packet) containing information about the registered server of the specified type that is most easily accessible according to the routing (network) table. This means that the server residing in the network with the best registered delay metric is chosen. If there is a tie for the best registered delay metric, the server registered with the least hops metric is chosen. If there is still a tie, the server that also occurs first on the server list is chosen. The chosen server is registered through the network from which the Nearest Service Query packet was received, or if any server exists in this network, then IPX Router will not respond to

this query packet (servers or other IPX routers in this network will respond). To establish a session with the nearest server, the client workstation usually chooses the IPX router (server) from which the Nearest Service Response was received first.

SPX Protocol

NetWare IPX (Internetwork Packet Exchange) and SPX (Sequenced Packet Exchange) represent two basic types of network communication protocols: IPX is connectionless and SPX is connection-oriented. Since IPX performs only the task of the Network layer of the OSI, it offers the benefits of speed and performance that result from its low overhead. However, IPX services are insufficient if the guarantees of the Transport OSI layer are needed. SPX is identical to IPX except that it has the additional overhead of the Transport OSI layer. These additional tasks make SPX a connection-oriented protocol. This means that before an SPX packet is sent, a connection between the sender and the receiver is established. SPX performs the tasks of guaranteeing delivery, sequencing packets, detecting and correcting errors, and duplicating packet suppression. These are the SPX benefits:

- *Guaranteed delivery* - Connection is established before sending information and returning verification (acknowledge packet) to the sender. The disadvantage is that if the receiver is not available, the packet cannot be sent. Broadcasting to multiple stations is not allowed because the connection must be established before sending information.
- *Guaranteed packet sequencing* - Regardless of how many packets a message requires it arrives in the proper order.
- *Duplicate packet suppression* - During the process of guaranteeing delivery (which improves resending packets presumed lost), it is possible for a duplicate packet to arrive at the receiving side. SPX discards such packets, so the application receives one copy of data from a communication partner.

SPX Packets

The SPX packet is identical to the IPX packet except that it has an additional 12 bytes SPX header. Figure 31 and Table 11 below describe the SPX header.

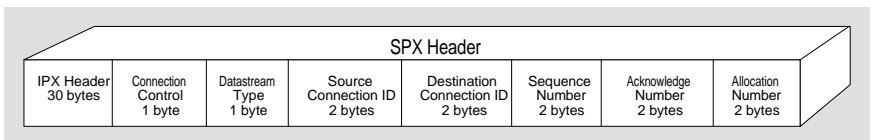


Figure 31. SPX Header

Field	Description
IPX Header	<p>Contains the standard 30-byte IPX header (see the description of IPX packets) with two SPX-specific restrictions:</p> <ul style="list-style-type: none"> • The Packet Type field is always set to 5 by SPX. • The Destination Node field never contains a broadcast address. <p>Each connection formed by SPX is with exactly one communication partner. Broadcasting or multi-casting is not allowed in SPX.</p>
Connection Control	<p>Contains four single-bit flags used by SPX and its clients to control the bidirectional flow of data across the connection:</p> <ul style="list-style-type: none"> • <i>Undefined</i> (1H-8H) - These bits are not defined by the Xerox Sequenced Packet Protocol. SPX ignores them. • <i>End-of-Message</i> (10H) - Client sets the flag to signal the end-of-connection to the partner (usually this bit is set in End-of-Connection-Acknowledgement packet - see “Datastream Type” below). SPX ignores this bit and passes it unchanged to the partner. • <i>Attention</i> (20H) - Client sets this flag if the packet is an attention packet. This feature has not been implemented. SPX ignores this bit and passes it unchanged to the partner. • <i>Acknowledgement Required</i> (40H)- SPX sets this bit if an acknowledgement packet is required. Since SPX handles acknowledgement requests and responses, clients don't have to be concerned with it (usually this bit is set in a <i>data</i> packet). • <i>System Packet</i> (80H) - SPX sets this bit if the packet is a system packet (e.g., acknowledge packet). These packets are used internally (do not need any acknowledge packets from the opposite side of the SPX connection) and are not delivered to the clients. <p>Clients should never use or modify the Undefined, Acknowledgement, or System bits. These are reserved for use by SPX.</p>

Table 11. SPX Header

Datastream Type	<p>This field is a 1-byte flag that indicates the type of data found in the packet. Possible values and definitions for the field are as follows:</p> <ul style="list-style-type: none"> • <i>Client Defined</i> (0H-FDH) - SPX ignores this field. • <i>End-of-Connection</i> (FEH) - When the client makes a call to terminate an active connection, SPX will generate an End-of-Connection packet (with the Acknowledgement Required bit set). This packet is then delivered to the connection partner as the last message during the connection. • <i>End-of-Connection-Acknowledgement</i> (FFH) - SPX generates an End-of-Connection-Acknowledgement packet automatically. This packet is not delivered to the partner clients.
Source Connection ID	Contains a connection identification number assigned by SPX at the packet's source.
Destination Connection ID	Contains a connection identification number assigned by SPX at the packet's destination. This field is used to demultiplex incoming packets from multiple connections arriving at the same socket. Demultiplexing is necessary because concurrently active connections on any machine may use the same socket number.
Sequence Number	This field keeps a count of packets exchanged in one direction on the connection. Each side of the connection keeps its own count. The number wraps to 0H after reaching FFFFH. Since SPX manages this field, client processes doesn't have to be concerned with it.
Acknowledge Number	This field indicates the Sequence Number of the next packet SPX expects to receive. Any packet with a Sequence Number less than the specified Acknowledge Number is in the correct sequence and need not be retransmitted. Since SPX manages this field, client processes need not be concerned with it.
Allocation Number	This field indicates the number of listen buffers outstanding in one direction on the connection. SPX may only send packets until the Sequence Number equals the remote Allocation Number. Since SPX manages this field, client processes need not be concerned with it.

Table 11. SPX Header

SPX Session Parameters

Parameters to control SPX sessions running on an IPX workstation or on a NetWare server can be set in NET.CFG on the IPX workstation or using SPXCONFIG.NLM on the NetWare server. Time related parameters are in tics (there are 18.21 tics per second on IBM PC compatibles). Some experiments show that time-related parameters are in fact from 1.5 to 2 times longer than the settings. For instance, when SPX Verify Timeout is set to 54 tics (about 3 sec.), the timeout is about 6 sec. Care must therefore be taken when setting time-related SPX parameters.

SPX session parameters are listed below by NetWare server parameter name with the parameter name on the IPX workstation in parentheses:

- **SPX Watchdog Abort Timeout** (SPX ABORT TIMEOUT)

This parameter adjusts the amount of time that SPX waits, without receiving any response from the other side of the connection, before it terminates the SPX session.

- **SPX Watchdog Verify Timeout** (SPX VERIFY TIMEOUT)

This parameter adjusts the frequency at which SPX sends a packet to the other side of the connection to inform it that its side is still alive. If no packets are exchanged on the SPX connection by the software that established the session, SPX will send keep-alive packets at regular intervals to make sure that the connection is still working. Each keep-alive packet is sent as an SPX acknowledge packet (i.e., System Packet - see the Connection Control field described earlier in the table). Sequence Number, Acknowledge Number, and Allocation Number are the same as in the last SPX session packet transmitted on this side of the SPX connection. Both sides of the SPX connection generate keep-alive packets independently.

- **SPX Ack Wait Timeout** (SPX LISTEN TIMEOUT)

This parameter adjusts the time that SPX waits, without receiving a packet from the other side of the connection, before it starts requesting that the other side send back a packet assuring the connection is still valid. If SPX does not hear from the other side of the connection during this time, it will send packets to the other side asking for verification that the connection still exists. Each verification packet is sent as a keep-alive packet with the Acknowledgement Required bit set. If this verification packet is received, a standard keep-alive packet is immediately sent back to the requester. Sequence Number, Acknowledge Number, and Allocation Number are not changed on either side during this operation.

- **Maximum Concurrent SPX Sessions (SPX CONNECTIONS)**

This parameter specifies the maximum number of SPX connections a workstation (or a server) can use at the same time.

- **SPX Default Retry Count (IPX RETRY COUNT)**

This parameter sets the maximum number of times a workstation or a server resends an SPX data packet.

SPX Spoofing

When an SPX session is running, keep-alive packets are sent periodically in both directions while no data is exchanged. This conveniently keeps the connection established, but the keep-alive packets can lead to unnecessary traffic forwarding through the router. This is important to consider when using point-to-point lines, especially Dial On Demand lines. In such cases, it is useful to be able to filter rather than forward unnecessary traffic and, if the line is otherwise unneeded, to temporarily disconnect the DOD line to save connection cost or reduce bandwidth used by an SPX session. SPX spoofing has been implemented to allow you to keep an SPX session running without causing unnecessary keep-alive traffic on point-to-point lines such as Dial On Demand connections. When spoofing is enabled, IPX router filters SPX keep-alive packets forwarded through the circuit and SPX keep-alive packets are sent back to the sender.

The SPX session direction on which transmitted SPX Keep-Alive packets and SPX Verification packets are filtered (due to SPX spoofing) is the *filter* direction. The opposite direction of the SPX session, on which SPX Keep-Alive packets are sent back to the sender to spoof an SPX station (placed on the opposite side of the point-to-point connection), is the *send-back* direction.

SPX sessions are registered by the router. They are automatically canceled when the router detects the end of the session or that the station on the send-back direction has become invisible.

SPX Spoofing Parameters

The following parameters should be set to run SPX Spoofing:

- **SPX Spoofing switch**

This parameter enables or disables SPX Spoofing on a selected point-to-point IPX circuit. This switch cannot be enabled on a LAN or IRL (ABC) IPX circuit.

- **SPX Session Termination Timeout**

This IPX router global parameter determines the minimum time to wait before starting to end an SPX session when no SPX packets are received. The maximum time is twice as long. If the Session Termination Timeout expires,

the End-Of-Connection SPX packet destined to the opposite side of the SPX connection is sent through the relevant point-to-point line and the SPX Spoofing mechanism waits for the End-Of-Connection-Acknowledge SPX packet for about 1/2 Idle Time (DOD parameter). If this packet is not received during this time, the whole procedure is repeated (maximum 10 times). If this acknowledge packet is received or is not received within 5 * Idle Time period, all structures associated with this SPX session are removed from this unit.

➔ **Note:** Set this parameter to about twice as long as the longest SPX Watchdog Abort Timeout (SPX ABORT TIMEOUT) parameter set for the SPX sessions running through this IPX Router unit.

Limitations

- SPX Spoofing requires IPX Router running on an XL unit.
- SPX Spoofing can trace SPX sessions running through point-to-point lines only.
- On an IPX circuit connected to a point-to-point line through which the *filter direction* is realized, the SPX Spoofing switch must be enabled. On an IPX circuit through which the *send-back direction* is realized, the SPX spoofing switch must be disabled.
- SPX Spoofing can trace SPX sessions running in both directions through the same XL unit (IPX router) only.
- In the *filter direction*, the IPX split traffic option must be disabled.
- In the illustration below, there is one SPX station in network A and another in network B, with an SPX session established between them through routers A and B. If router A knows network B by two or more routes with the same IPX cost, router A must have SPX Spoofing enabled on all filter direction circuits connected with these routes. The same applies, in reverse, to router B: if it knows multiple ways to network A, SPX spoofing must be enabled on the filter

direction circuits associated with all of those routes.

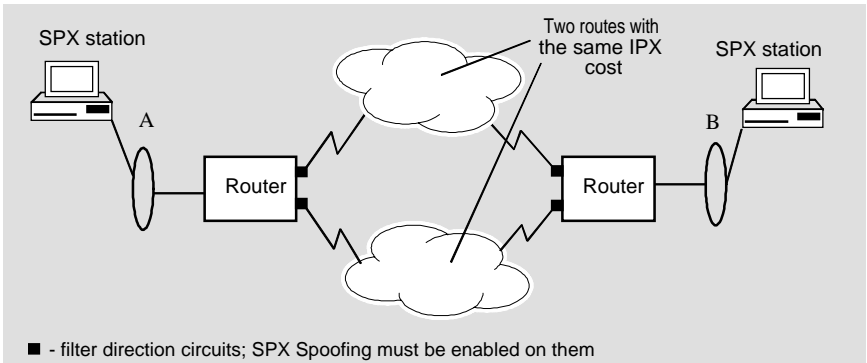


Figure 32. SPX Spoofing with duplicate routes with the same IPX cost

SPX Sessions and IPX Default Route

Figure 33 shows a sample configuration in which SPX Spoofing can be used with IPX Default Route. When the user wants to use an SPX session with default route, then he has to have IPXRTR.NLM (released in February 1995 or later) installed on IPX servers. This module has a Default Route support.

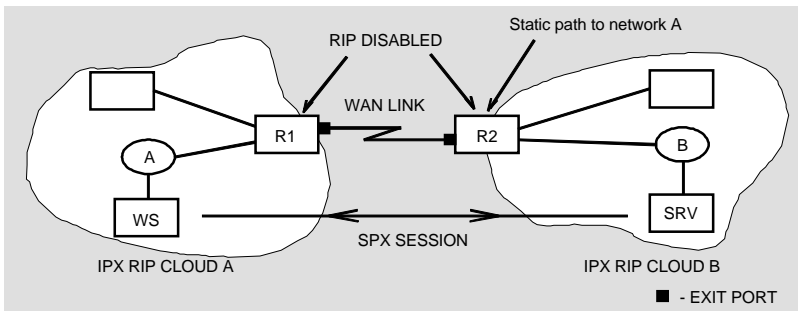


Figure 33. SPX Spoofing between two clouds in Default Route mode

If the new IPXRTR.NLM is unavailable, then the following method may be used. When the Default Route mode is turned on, then an IPX workstation is able to attach to an IPX server ver. 3.11, even if this server does not know a network on which the workstation exists. Unfortunately when the workstation tries to connect to a server using SPX packets, then the server does not respond until it registers this network.

This prevents from using e.g. Remote Console (RCONSOLE.EXE), running in one of the RIP clouds, to manage an IPX server in a different RIP cloud connected via Exit Router.

To work around this problem you can define in an Exit Router (in router R2 on the Figure 33) a static route to the network on which the workstation exists. Such network will be advertised in its own RIP cloud and an IPX server will register this network. This solves the problem for all workstations on this network.

SPX Spoofing Network Considerations

The following pictures show example SPX Spoofing configurations.

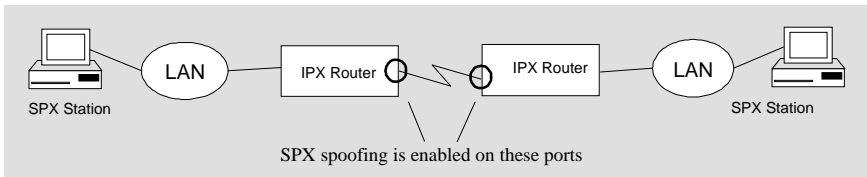


Figure 34. Basic SPX Spoofing Configuration

Figure 34 shows the basic configuration, where an SPX session is established between two SPX stations. The session is registered in both IPX routers and spoofed on selected IPX ports.

Figure 35 shows two IPX subnetworks separated by a network. SPX spoofing should be enabled on selected IPX ports to work properly.

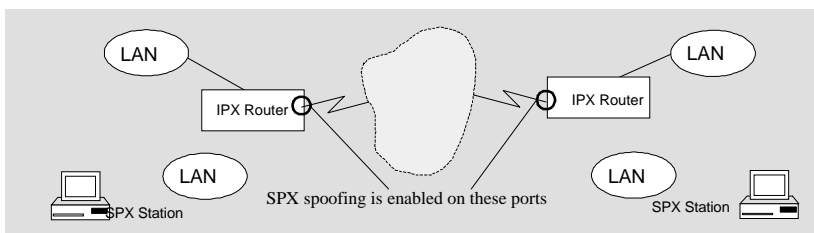


Figure 35. SPX Spoofing Configuration within a few LANs

Figure 36 shows a composite IPX network. There are three possible SPX routes realized through the point-to-point lines:

- LAN 1 - LAN 2
- LAN 2 - LAN 3
- LAN 1 - LAN 3 (IPX router 2 doesn't register SPX session between stations 1

and 3)

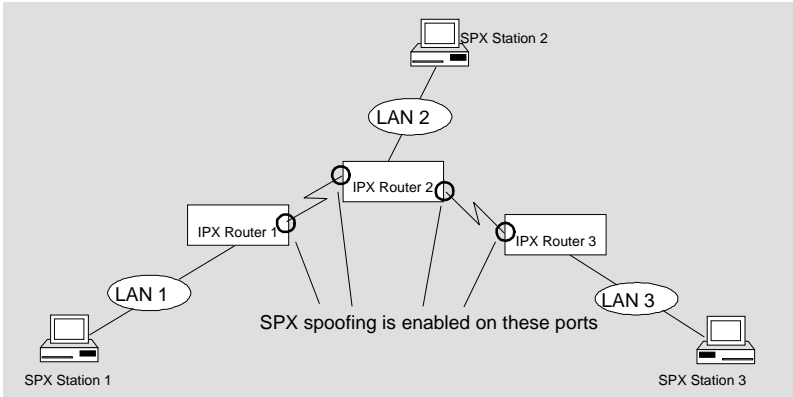


Figure 36. Composite IPX Network

SPX Spoofing Topology Restrictions

SPX spoofing should be set only on point-to-point lines. If SPX spoofing is enabled or disabled on one side of a point-to-point line, the same state must be set (enable or disable) on the other side of this connection.

IPX Default Route

NetWare has a reserved IPX network number that represents a **default route** (0xFFFFF0FE). This is an advertised destination to which IPX packets with unknown destination networks are forwarded. The remote router is instructed to send any packets addressed to an unknown network address to the router advertising the IPX default route. It is assumed that the router advertising the default route knows how to reach any destination.

➔ **Note:** Support for IPX default route was added to IPX with all versions of Novell's IPXRTR.NLM released in February 1995 or later.

Third party routers may be set up to advertise only the default route, with no other IPX network information. A NetWare server on the network will install this default route, but have no other IPX network information. As a result, all packets from the NetWare server will be forwarded to the specified default router.

This scenario may work properly as far as routing goes, but causes problems with SAPs (i.e., advertised servers). When a router or server receives a SAP broadcast, it checks its routing table to see if the advertised network number on which the advertised server exists is in its routing table. If it finds the network number, it adds the advertised server to its tables. If it does not find the network number, it discards this SAP entry. If the server's routing table consists of only the default

route, it will not see the advertised server's specific network number and discard the SAP entries. As a result, no SAP information will be retained.

Novell's MultiProtocol Router 3.1 has a new set command to get around this problem.

Syntax: SET REQUIRED NETWORK FOR SERVICES=OFF

If this is set, the NetWare server (or router) will not discard the SAP entries (with advertised server) packet if it does not find the network number in its routing table. Instead, it will check further to see if the server/router has a default route and, if so, it will add the SAP information to its tables. If neither the specific network nor the default route exists, the entry will be discarded.

Definitions

Name	Description
Default Route Mechanism	A mechanism for routing packets between two or more IPX RIP clouds when RIP advertising between these clouds is disabled or limited. One cloud can send a packet to the other cloud even if the specific IPX network number is unknown.
Learning Netless Servers	A mechanism for learning servers when an advertised server's network number is unknown to the IPX router. This mechanism can be active only when the default route mechanism is enabled.
Default Route	The path (or paths) to the default net (see below), which is used in the default route mechanism. The default route is used for routing to an unknown destination. This network always has the number FFFFFFFE and it is assumed that customers avoid using this number in their networks. In this document, the "default route" is network FFFFFFFE as learned by IPX router through an enabled IPX default route mechanism. Otherwise if the network FFFFFFFE is learned but the IPX default route mechanism is disabled, then this network is a regular network.
Default Net	Network number FFFFFFFE when IPX default routing is active.

RIP Cloud	An IPX network with at least one IPX router where the IPX network advertisement is based on RIP. A RIP cloud can be connected to another RIP cloud, usually via a LAN or WAN link. On this link RIP and SAP can be disabled, so a cloud may not know nets or servers from the other cloud via RIPs or SAPs. Selected networks and servers from another RIP cloud can be set as static ones in the local RIP cloud. The goal of such clouds is to reduce the advertising on a link between clouds and, for example, make it possible to define locally the same network numbers in separate clouds.
Exit Router	An IPX router placed on the edge of an IPX RIP cloud. In this router the default route is set as static on a port exiting the RIP cloud. Usually there is only one exit router per RIP cloud. Using more than one exit router is possible, but they must all connect the local cloud with exactly one remote cloud.
Exit Port	The port in an exit router through which the default route is registered. The next hop in this default route should be the remote router through which the remote RIP cloud (or clouds) can be accessed.
Default Route Mode	If IPX is in default route mode, it can route packets between two or more IPX RIP clouds when RIP advertising between these clouds is disabled or limited.
Learning Netless Servers mode	If IPX is in learning netless servers mode, it is in default route mode and it can learn servers when an advertised server's network number is unknown to the IPX router.

When IPX is in default route mode, it registers a path (or paths) to the default net (FFFFFFFE) as a default route. Now if an IPX packet is received and the destination net is unknown, the router sends this packet to the port on which the default route is known. If the default route is unknown, then the packet is dropped. The default route can be registered from RIP or set statically. In exit routers the default route must be set statically.



Note: If default routing is used, the net FFFFFFFE cannot be set anywhere as a real net.

IPX Default Route Network Considerations

Sample configurations:

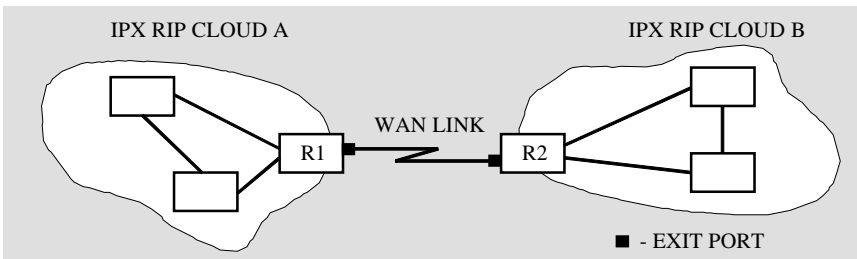


Figure 37. Sample basic configuration

The above picture shows two IPX RIP clouds connected via WAN link. Both clouds are connected with each other via exit routers R1 and R2. In exit router R1, the next hop for the default route is router R2, and in exit router R2, the next hop for the default route is router R1. Exit ports are the ports connected to the WAN link in exit routers R1 and R2. A packet generated in cloud A to an unknown IPX net will be sent to cloud B via exit router R1. Exit router R2 receives the packet and forwards it to a port on which the destination net is registered.

If a packet is received on an exit port and the exit router (e.g., R2) does not know the destination net, then the packet will be sent back to the RIP cloud that sent the packet (in this example this packet will be sent back to router R1).

To avoid this behavior, the **IPX Backward Routing** option should be disabled on exit ports. If this option is enabled (default setting) then the packet will go from one exit router to another until the transport control field in the IPX header reaches 16 hops and the packet is dropped. This situation is possible if IPX networks in RIP clouds are not stable, i.e., if an IPX port in one cloud is coming up and down, then some nets may be periodically learned and timed-out.

Other legal configurations:

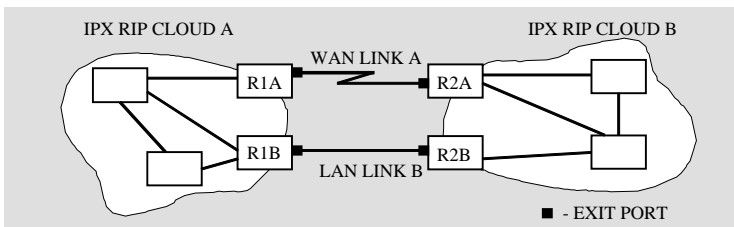


Figure 38. Two IPX RIP clouds connected via two redundant links and four exit routers

Two clouds can be connected via two links (WAN or LAN), with each link in each cloud on a different exit router. If one of the links or routers fails then other link will keep the connection between clouds. If one of the links is preferred by the user, the better route to the default net should be set on the exit port of the preferred link.

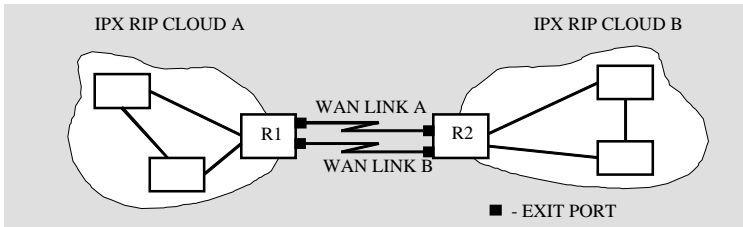


Figure 39. Two IPX RIP clouds connected via redundant WAN links on two exit routers

Two clouds can be connected via two links (WAN or LAN), with two links in each cloud on the same exit router. If one of the links fails then other link will keep the connection between the clouds. If one of the links is preferred by the user then the better route to the default net should be set on the exit port of the preferred link. If one exit router fails the connection will be lost.

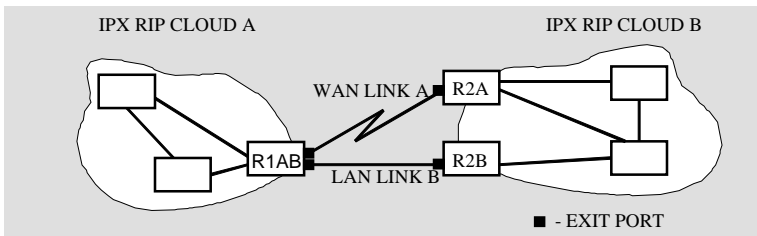


Figure 40. Two IPX RIP clouds connected via redundant WAN links on three exit routers

Two clouds can be connected via two links (WAN or LAN) involving three exit routers. In cloud A, two links are on the same exit router. In cloud B, the two links are on different exit routers. If one of the links fails then other link will maintain the connection between clouds. If one of the links is preferred by the user, the better route to the default net should be set on exit port of the preferred link. If exit router R1AB fails, the connection between clouds will be lost. In cloud B, however, the failure of one exit routers will not break the connection between clouds.

Improper configuration:

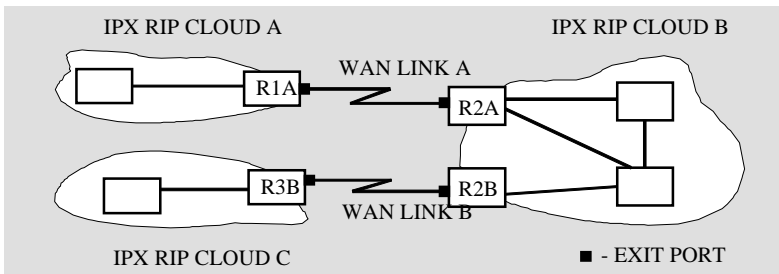


Figure 41. Three IPX RIP clouds connected improperly

This configuration is wrong and the default route mechanism will not work because cloud B has exit routers connected to two different clouds.

More elaborate configurations:

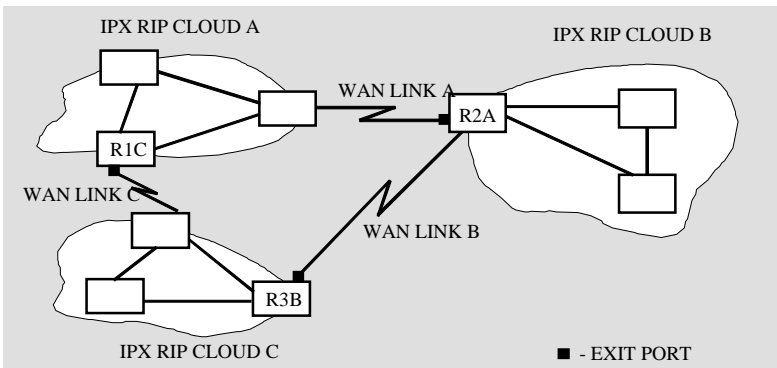


Figure 42. Three clouds connected via default route

You can connect three IPX RIP clouds via three WAN (as above) or LAN links.

In this example, exit router R1C sends IPX packets with an unknown destination net to cloud C. If cloud C knows the destination net, it sends them to the proper router in cloud C. Otherwise it forwards these packets to cloud B.

If cloud B knows the destination net, it sends the packets to the proper router in cloud B. Otherwise it sends these packets back to cloud A. A packet could be transmitted in such a loop until it ages out (hop count = 16).

This configuration is legal, but not typical, and default routing will work, but each IPX session between the two clouds will always use three links. The fail of one link will break the connections between the three clouds. Links between clouds

can be duplicated to achieve better fault tolerance. Of course, for example, a single LAN segment can be used instead of these three WAN links to have another legal configuration.

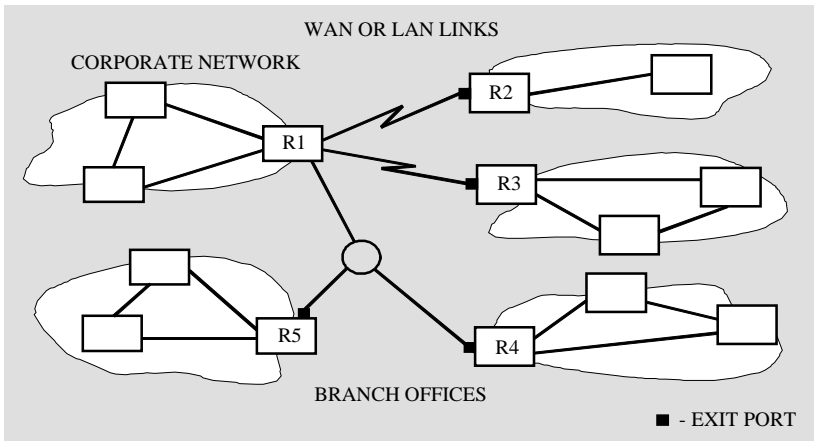


Figure 43. Branch offices connected to headquarters via default route

Router R1 (at headquarters) should have RIPs and SAPs blocked on ports connected to branch offices. RIP and SAP blocking is performed using IPX SAP and RIP filters for transmitted RIP and SAP packets. WAN links will then not be loaded with RIP/SAP control traffic from the corporate network. Exit routers R2, R3, R4, and R5 have the default route and some servers from the corporate network set statically, and RIPs and SAPs are enabled on their exit ports (this is different from previously discussed configurations). With this configuration, IPX workstations from the branch offices can connect to selected servers from the corporate network. IPX routers in the corporate network register networks and servers from branch offices, because branch offices advertise themselves. The RIP/SAP advertised traffic from branch offices is very low because they are relatively small networks, usually just one router and some workstations. On WAN links reduced advertising can be used to reduce RIP/SAP advertisement traffic from branch offices more effectively. Alternately, depending on the desired configuration, RIPs and SAPs can be disabled completely on WAN links, and static networks and servers can be used instead.

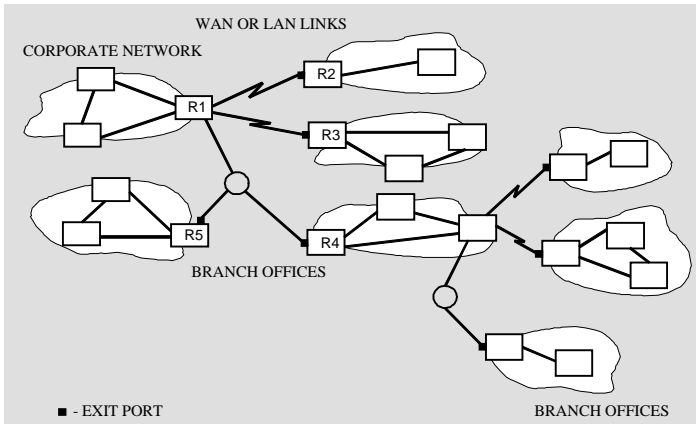


Figure 44. Branch offices connected to headquarters via default route

It is possible to extend the previous configuration by connecting more branch offices through existing branch offices to the corporate network. When another branch office is connected to an existing branch office, the additional branch office should be configured following the same procedure as described before. Then the new branch office treats the existing branch office as the corporate network.

Duplicated network numbers in different RIP clouds

If RIP advertising is disabled or limited (filtered) on exit ports, it is possible to define duplicate network numbers in both RIP clouds connected via exit routers in default route mode.

An IPX workstation on network AAA in Cloud A can connect to the server on network BBB in cloud A but can not connect to the server registered on network BBB in cloud B.

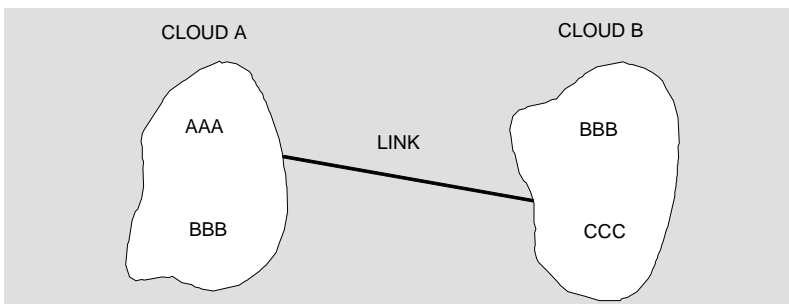


Figure 45. Duplicated network numbers in different RIP clouds

An attempt to connect from network BBB in cloud A to network CCC in cloud B will fail: a packet sent from network BBB in cloud A to the IPX server on network CCC in cloud B will go through the link between clouds, and then server on network CCC in cloud B will try to send an answer packet to originating workstation on network BBB in cloud A, but this packet will be routed to network BBB in cloud B.

A similar sequence occurs when sending a packet from an IPX workstation on network CCC in Cloud B to the server on network BBB in cloud B. The only possible connection between the two clouds is from network AAA in cloud A to network CCC in cloud B.

- As a rule, both networks participating in the IPX session must have unique numbers.

Server Advertising

When using default route mode, server advertisement can be disabled or limited (using SAP filters) on links between RIP clouds. Then no SAP information or reduced SAP information can be propagated between clouds. Each IPX router in RIP clouds must be set to learn netless server.

When an IPX workstation sends a “get nearest server” request to its local server or IPX router, this server or router has to know a server from the other cloud to be able to make a connection.

To make this work, you could set static remote servers in the local exit router or set appropriate SAP output filters in the remote exit router. The local exit router will propagate SAP information about selected servers known from the remote RIP cloud inside its own cloud.

Another way is to attach to a single server from the remote RIP cloud, which makes it possible to log in to another remote server. All necessary information about logging in will be provided by the server currently attached. In such cases it is enough to set statically only one server from the remote RIP Cloud.

Network Considerations

This section includes an example of a network using IPX routing as well as topology restrictions.

Configuration Example

IPX Routing provides internetwork integrity and stability, reduce unnecessary traffic and ultimately increase the throughput within an IPX internetwork. These benefits are achieved by a proper connection that is characterized by:

- *Transmission Medium Independence* -- all connections between different types of network interfaces are possible.
- *Local Traffic not Forwarded* -- IPX local traffic is not forwarded to other segments.
- *Packets Forwarded over Efficient Paths* -- all IPX packets addressed to remote segments (non-local segments) are delivered by way of the most efficient path.
- *IPX broadcast packets are not flooded* -- RIP/SAP broadcast packets are not forwarded to other segments. Also, propagation of other IPX broadcast packets (for example, NetBIOS over IPX) is reduced.
- *Connection Failures are Automatically Detected for Fast Switching to an Alternative Path.*

You must decide which routers in an internetwork should be enabled with IPX Routing to achieve the greatest benefit. The speed of the networks and devices is a major consideration. Despite the fact that an IPX router can be a bit slower than a Non-IPX router (regular bridge), due to the benefits mentioned above, enabling IPX Routing usually results in greater efficiency. Consider the following example.

In Figure 46, if a workstation in Group 1 wants to communicate with File Server 2 through Router 1 and Router 2, and Router 1 and Router 2 are operating as bridges (not IPX routers), the workstation views the network segments between Ring 1 and Ring 2 as a single IPX network segment. This means that some local traffic from Ring 1 or Ring 2 (for example, broadcast packets or some session initialization packets) will be forwarded through the slow (WAN) line. This could degrade the WAN line's performance, cause some Group 1 workstations to lose connections with File Server 2, and slow down remaining sessions. It might also become impossible to switch fast existing sessions between Group 1 and File Server 2 to the alternate path (through Router 1, Router 4, Router 5, Router 6) when the WAN line fails. If, however, Router 1 and Router 2 were configured as IPX routers, the workstations in Group 1 would view the path from one Token Ring to the other through the WAN as a complete path: Ring 1, WAN, and Ring 2 would be treated as separate IPX network segments, local traffic would not be forwarded through the WAN line, IPX broadcast packets (for example, RIP and

SAP packets) would be terminated by Router 1 and Router 2, and connections would not be broken or slowed down.

Router 4 and Router 5 should be configured for IPX Routing so that the connection between the two Ethernet segments through the X.25 interface is a complete path.

Router 3 can operate as a bridge or IPX router depending on the traffic intensity in Token Ring segments R2 and R3. If the traffic is light, Router 3 can operate as a bridge.

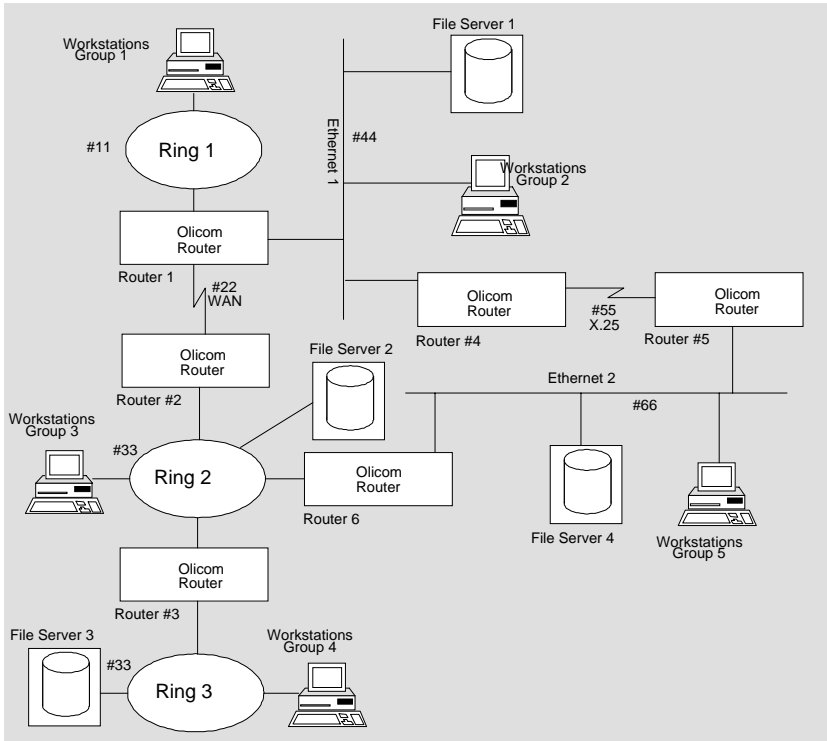


Figure 46. IPX Configuration Example

While Router 6 can be replaced by a direct connection from File Server 2 to the Ethernet 2 Segment, a file server should not be used as an IPX router; a router provides a faster access time for the workstations especially if the File Server 2 is heavily used.

In Figure 46, Routers 1, 2, 4, 5 and 6 have IPX Routing enabled and Router 3 does not. The IPX network numbers are set for each port (IPX circuit) connected to the segment.

Advertising (RIP/SAP) traffic generated by the IPX routers can be reduced by:

- Setting Reduced Advertising on ports (IPX circuits) connected to the X.25 line (on Router 4 and Router 5) and on ports connected to a WAN line (on Router 1 and Router 2).
- Setting both: Disable RIP/Disable SAP packet processing on each end of the X.25 line (Router 4 and Router 5) AND static networks/static servers for all networks and servers registered through the X.25 line. Do not set static networks and static servers for the WAN line because Router 1 is a single router connected to Ring 1 that should register two paths for some networks and servers (Router 1 is also connected to Ethernet 1 and the WAN line).

If a speed of 9600 bps is set on the X.25 line (delay = 11) and if File Server 1 is assigned a name FS_1, an internal network number of 444 is assigned and the physical port address of Router 4 is 16-00-98-40-11-12. If the X.25 line connects to Port 5 (port 5, circuit 1) on Router 5, then the following static network and static server can be set on Router 5:

- IPX network number: 444, Hops: 2, Delay:13, Port:5 (port 5, circuit 1), Next Hop MAC Address: 16-00-98-40-11-12
- Server name: FS_1, Server type: 04, Server net number: 444, Server node (MAC) address: 00-00-00-00-00-01, Server socket: 451.

- Increasing Update Interval and Maximum Packet Size for RIP and SAP protocols on ports connected to the X.25 line (Router 4 and Router 5) and the WAN line (Router 1 and Router 2).

Other IPX traffic forwarded through the slow lines can be reduced by: *setting IPX NetBIOS reduced propagation* on all routers or setting *IPX watchdog spoofing* at each end of the WAN line and X.25 line.

An additional advantage is achieved by enabling IPX Routing as indicated in Figure 46. Each of the following connections has two paths:

- Between Ring 1, Ring 2, and Ring 3.
- Between Ring 1 and Ethernet 2.
- Between Ethernet 1, Ring 2 and Ring 3.
- Between Ethernet 1 and Ethernet 2.

One path is through the WAN segment and the other through the X.25 segment. If one path fails, the connection automatically switches to the alternate path. The time required to switch to the alternate path depends on the reason for the failure. When an interface fails and the active state is lost, the path switches within a few

seconds. However, when an interface fails but no state change occurs, it takes up to four minutes for the path to switch.

To decrease the time for IPX traffic to switch from a path through Router 1 to a path through Router 4, the Age Multiplier for the RIP and SAP protocols can be decreased on ports connected to the WAN line (on Router 1 and Router 2).

If File Server 1 does not exist, then a connection can be forced from the workstation in Group 2 to a preferred server through Router 4 by setting Enable/Disable SAP Get Nearest Server Reply on the port (IPX circuit) connected to Ethernet 1 segment. To force a preferred server connection through Router 1 or Router 4, enable SAP Get Nearest Server Reply on the Ethernet port (IPX circuit) connected to Router 1 and disable SAP Get Nearest Server Reply on the Ethernet port (IPX circuit) connected to Router 4.

By setting relevant network filters or server filters, you can control access from relevant segments to relevant servers and networks, reduce the number of registered paths for selected networks and servers, and force connections through selected paths.

If both of the following are true:

- File Server 1 is named FS_1, the internal network number is 444, and Router 5 connects to the Ethernet 2 segment through port 1 (port 1, circuit 1) and through port 2 (port 2, circuit 1) to the X.25 line.
- Router 6 connects to the Ethernet 2 segment through port 1 (port 1, circuit 1) and through port 2 (port 2, circuit 1) to Ring 2.

Then, you can set the following network or server filters to disallow access from Workstation Group 5 to File Server 1:

- Network filters -- IPX network number 444, Output Positive Filter on port 1 (port 1, circuit 1) Filter net if RIP update sent to port 1 (port 1, circuit 1) or Input Positive Filter on port 2 (port 2, circuit 1). Filter net if RIP update received on port 2 (port 2,circuit 1). Set the same filters on Routers 5 and 6.
- Server filters -- Server name is FS_1, Server type is 04, Server node address is 00-00-00-00-00-01, Server network number is 444, Server socket is 451, Output Positive Filter on port 1 (port 1, circuit 1). Filter server if SAP update sent to port 1 (port 1, circuit 1) or Input Positive Filter on port 2 (port2, circuit 1). Filter server if SAP update received on port 2 (port 2, circuit 1). Set the same filters on Routers 5 and 6.

To allow access to File Server 1 only from Ethernet 2 segment, set the following:

- Network filters -- IPX network number is 444, Output Negative Filter on port 1 (port 1, circuit 1) Filter other networks if RIP update sent to port 1 (port1, circuit 1) or Input Negative Filter on port 2 (port 2, circuit 1) Filter other

networks if RIP update received on port 2 (port 2, circuit 1). Set the same filters on Routers 5 and 6.

- Server filters -- Server name is FS_1, Server type is 04, Server node (MAC) address is 00-00-00-00-00-01, Server network number is 444, Server socket is 451, Output Negative Filter on port 1 (port 1, circuit 1) Filter other servers if SAP update sent to port 1 (port 1, circuit 1) or Input Negative Filter on port 2 (port 2, circuit 1) Filter other servers if SAP update received on port 2 (port 2, circuit 1). Set the same filters on Routers 5 and 6.

Topology Restrictions

IPX Routing can be applied to any network topology by following these configuration rules.

WAN, X.25, Frame Relay, ISDN, and ATM Point-to-Point Lines

For WAN, X.25, Frame Relay, ISDN, and ATM point-to-point lines, these configuration rules apply:

- If IPX Routing is enabled on one side of the point-to-point line, it must be enabled on the other side of the point-to-point line.
- If Reduced Advertising is enabled or disabled on one side of the point-to-point line then the same configuration option (enabled or disabled) must be set on the other side of the point-to-point line. Also, be sure to set the same type of Reduced Advertising (*Unacknowledged* or *Acknowledged*) on both ends of the point-to-point line. The *Unacknowledged* type is compatible with Reduced (Smart) Advertising (or Auto option) used in all ILAN and XL versions prior to XL 4.1.
- If RIP (or SAP) packet processing is enabled or disabled on one side of the point-to-point line, then use the same setting on the other side. Also, you must set all RIP (or SAP) packet parameters to identical values on both sides of the point-to-point line.
- The same Port Type (IPX Circuit Type), with WAN Numbered RIP or WAN Unnumbered RIP, must be set on both sides of the point-to-point line. Also, the same unique IPX network number must be assigned to the port (IPX Circuit) connected to the point-to-point line if WAN Numbered RIP or Broadcast port (IPX circuit) is used.
- The same IPX Packet Format must be set on both sides of the point-to-point line. Note that in all ILAN and XL versions prior to XL 4.1, Ethernet 802.3 IPX Packet Format was only used on point-to-point lines.

SRT or SR Mode

If source routing is enabled on the IPX router port connected to Ring 1 and a bridge is operating in Source Routing or Source Routing Transparent mode, a file server must have a Source Routing driver (ROUTE.NLM or ROUTE.VAP) installed. For an Olicom IPX router, source routing mode can be disabled on the port connected to Ring 1 when the bridge is operating in SRT mode. By disabling the IPX SR option, you do not need for a Source Routing driver installed on the file server. Refer to the “ClearSight IPX Configuration” section in online for more information.

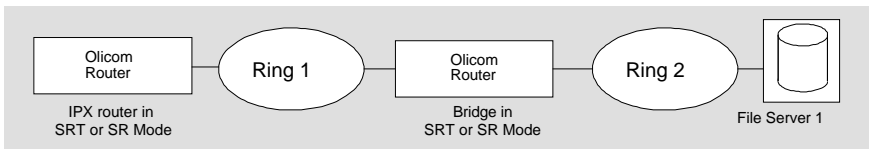


Figure 47. IPX Router in SR or SRT Mode



8. Protocol Independent Routing (PIR)

This chapter explains the features and implementation of Olicom's Protocol Independent Routing (PIR).

Sections

- *Overview*
- *Technical Discussion*
- *Network Considerations*

Overview

PIR is a state-of-the-art routing technique for LAN internetworking. PIR provides a method of routing LAN packets through complex mesh networks without the use of Network-Layer (Level 3) addresses. PIR implements the Address Processor and Directory on the Olicom router to provide routing capability based on the MAC-layer information in the Token Ring or Ethernet address of every packet.

The ability to route packets without using a Network-Layer Address (NLA) is significant because a many protocols (like SNA and NETBIOS) do not include Network Layer Addressing in their protocol information. PIR routes protocols that do have NLAs such as Novell IPX, IP, and DECnet. However, the network addresses of the PIR-connected LANs must be identical.

The routing algorithm implemented in an Olicom router running PIR is DSPF (Discovery- Shortest Path First). This algorithm automatically and dynamically chooses the best path for a packet traveling in a meshed network. In addition, DSPF is an adaptive routing protocol that periodically verifies and changes routing tables based on traffic conditions within the network.

The DSPF routing algorithm has the following abilities:

- *Operates with Source Routing and non-Source-Routing end-stations* -- A set of PIRs and network connections (LANs and WANs) that are used to interconnect PIRs is referred to as a *PIR cloud*. A PIR cloud appears to external stations as a single Source Routing network. End-stations that use Source Routing and that depend on the Source Routing field to adjust session timers operate correctly with PIR. By making the cloud appear as a single (virtual) hop, an Olicom router operating with PIR can dynamically choose paths within the cloud without having an impact on the end-stations.

- *Effectively uses all WAN interfaces in a mesh network* -- When using PIR, all WAN interfaces are available for use by the routing algorithm. No links are kept in a blocking or standby state. The routing algorithm dynamically chooses the best path between each pair of LANs or WAN segments. (The choice may not be the same in both directions.)
- *Dynamically adapts the routing paths to network loading and congestion* -- The Olicom router periodically measures the delay across available paths and changes the routing tables to reflect the lower delay paths if they are available. Through a number of unique aspects of the DSPF algorithm, this is done in such a way that packets are not delivered out of sequence due to path changes.
- *Quickly and transparently recovers from interface and device failures* -- The DSPF algorithm allows very fast recovery and convergence after various types of failures (for example, WAN interface dropping, Token Ring media problems, unit failures, etc.). If alternative paths exist, they can be put into service (transparently to end-stations) in as little as one to two seconds depending on the maximum diameter of the network.

PIR provides all of the benefits of contemporary routing protocols to protocols that do not support a native dynamic routing architecture.

Technical Discussion

The section discussed the following PIR terminology and concepts:

- DSPF Basics
- DSPF/Source Routing Interactions
- Selecting the DSPF Version
- Source Routing and LAN Segment Addressing
- PIR in a Source Routing Environment
- PIR Areas

DSPF Basics

The DSPF (Discovery Shortest Path First) protocol implemented in PIR is based on many of the same techniques as the link state Open Shortest Path First (OSPF) Protocol. Performing tasks at the Data Link level, an Olicom router running PIR creates a Virtual Network level and is responsible for routing packets through the network (PIR's cloud). The PIR delivers packets by way of the most efficient path. The determination of the best path between LAN segments is made independently for each direction and need not be the same in both directions. The criteria for the most efficient path is based on time as well as the current load level on a particular link.

An Olicom router running PIR creates a backup path if a primary path fails. This prevents losing sessions between end stations. Additionally, the DSPF algorithm avoids temporary loops in the PIR network and adjusts to eliminate missequencing errors.

DSPF/Source Routing Interactions

PIR provides the following methods for dealing with Source Routing frames:

- DSPF/T (Transparent routing)
- DSPF/SR (Source Routing only)
- DSPF/SRT (Source Routing and Transparent routing)

In these methods, PIR associates a LAN segment with every MAC address. The methods differ in how they deal with Source Routing discovery packets (explorer frames) and how the Olicom router updates the Routing Information Field (RIF).

DSPF/T

In DSPF/T mode, an Olicom router does not process, modify, or interpret the RIF. All user data messages are transmitted transparently through the PIR network, regardless of whether they contain Source Routing information or not.

DSPF/SR

In DSPF/SR mode, only Source Routing frames are forwarded by the Olicom router. Non-Source Routing frames are filtered. The entire PIR cloud is a virtual segment and a single hop in the RIF. Source Routing Explorer frames that traverse the PIR cloud reach their destination with two Routing Descriptors added to their RIF fields. The first is the incoming boundary LAN to the virtual segment and the second is the virtual segment to the outgoing boundary LAN. You can specify the ring numbers associated with these virtual segments.

DSPF/SRT

DSPF/SRT is the default mode for an Olicom router running PIR. Source Routed frames are forwarded as described above for DSPF/SR, and non-Source Routed frames are forwarded transparently.

Selecting the DSPF Version

DSPF has the following working modes:

- *Off* --DSPF paths are not built but management traffic is delivered using bridging.
- *Compatibility* --This mode supports ILAN PIR only. In this mode PIR chooses Version 1 or Version 2 of DSPF protocol based on the network configuration. If at least one PIR running Version 1 of DSPF is detected, then PIR begins using DSPF Version 1. Otherwise, PIR uses DSPF Version 2 but sends RESET messages in Compatibility format and always recognizes RESET messages in this format. This is an adaptive mode that continues to work properly in old mode until the last router in the network is updated with the new PIR software. This mode is chosen when upgrading PIR software from version 1 to version 2A. DSPF Compatibility Mode offers you two ways to upgrade your network from PIR version 1 to PIR version 2:
 - You can upgrade all PIRs to Version 2 gradually by configuring newly added PIRs in Compatibility Mode and upgrading existing PIRs with new software. When you upgrade the last PIR in your system, you can take full advantage of all Version 2 features.

- You can decide to keep old PIRs in a separate area of the network. All old PIRs should be kept in one area (given the same Area Number) and that area should be reserved strictly for old PIRs. The point is that old and new PIRs cannot be connected over a WAN line. New PIRs can be configured for several different areas and should be configured to run in Version 2 mode.
- *Version 1* --In this mode PIR uses the old versions of the DSPF protocol and frame formats for service messages. This mode does not recognize DSPF messages in version 2 format. Therefore, Compatibility Mode RESET messages are recognized but are answered with a RESET message in the version 1 format. Management traffic and ARP packets are still bridged within PIR cloud. New features like areas cannot be used in Version 1. This mode is totally compatible with previous PIR releases. This mode in ILAN does not support WAN connections to other Olicom routers which do *not* run PIR.
- *Version 2* --This mode is supported in ILAN PIR only. In this mode PIR only uses the new version of the DSPF protocol and the new frame formats for service messages. This mode does not recognize DSPF messages in version 1 format. Management traffic and ARP packets are delivered using the appropriate DSPF paths.

Source Routing and LAN Segment Addressing

Source Routing is processed in a special way in a PIR network. The entire PIR cloud appears as a single virtual LAN segment. This virtual segment allows for a change in path inside the cloud to improve efficiency or respond to a failure without any impact on the end stations.

Two separate tasks occur when a creating a virtual network. In the background the Olicom router running PIR creates paths to access every LAN segment in the PIR network, and in the foreground the Olicom router associates every known MAC address with the segment it was first detected on. Messages destined for a known address are transmitted to the associated segment through the selected optimal path. Messages addressed to unknown destinations are sent to all segments. The Olicom router dynamically handles the learning and aging of station addresses.

PIR in a Source Routing Environment

PIR has been designed to be compatible with and to cooperate with Source Routing bridges that connect LAN segments together beyond the border of the PIR mesh. All Source Routed frames (including discovery packets and specifically routed frames) can be forwarded throughout the PIR cloud to and from LAN segments that are internal to the cloud as well as external to the cloud.

Because the entire PIR cloud appears as a single hop, the number of actual hops that a source routed packet can traverse is greater than the maximum hop count limit of seven in ILAN and thirteen in XL.

In addition, because the PIR mesh is just a virtual hop, the actual route taken through the mesh is invisible to the end-stations. Thus, in the event of an interface failure, the transmission path can change without impact to the end-stations. In most cases, if an alternate path exists, the path change can occur in just a few seconds without session termination.

PIR Areas

A PIR area is one or more PIRs with the same Area Number. The separation of PIRs into areas is a logical (not physical) separation. Areas allow PIR networks to expand by creating groups of PIRs as separate PIR areas (or clouds). The DSPF protocol runs separately in each area limiting the number of protocol frames required to build routing tables.

The following are limits of area processing:

- Global, inter-area connections only form a tree topologies. No loops or duplicate connections between areas are allowed.
- If you run in DSPF/SR or DSPF/SRT mode, the RIF field restricts the topology to a seven-hop limit in ILAN and thirteen-hop in XL. The only topology allowed is a central area with connected satellite areas.

Network Considerations

This section includes the following topics:

- PIR Version 1 Configuration Example
- PIR Version 2A Configuration Example
- Topology Restrictions

PIR Version 1 Configuration Example

Figure 48 shows a typical mesh topology. The PIR cloud includes PIRs and the LANs and WANs used to interconnect them. The stations external to the cloud can communicate with stations internal to the cloud.

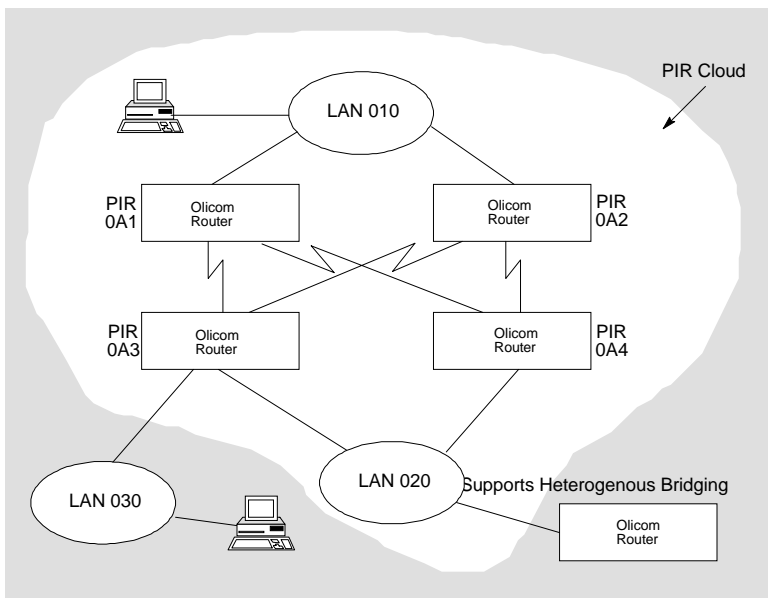


Figure 48. PIR Configuration Example

PIR Version 2A Configuration Example

When running PIR version 2A with multiple areas and using Source Routing, two hops are written in RIF field for each traversed area. The following elements are included:

- Segment number within the area through which the frame enters.
- Bridge number of the PIR cloud through which the frame enters.

- Bridge number of the area.
- Bridge number of the PIR cloud through which the frame exits.
- Segment number within the area through which the frame exits.

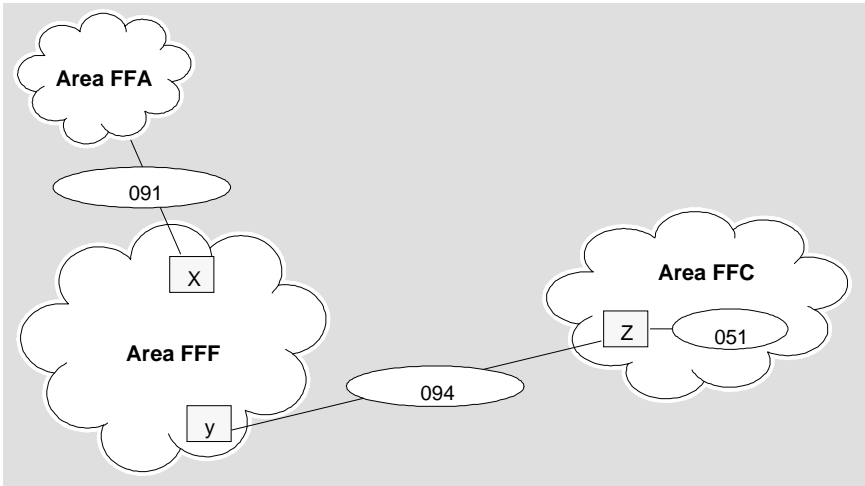


Figure 49. Central Area with Connected Satellite Areas

Because the RIF field is limited to seven hops, then a configuration must only consist of a central area with connected satellite areas. Therefore, the RIF field for frames going from one satellite area to another is six hops for frames going from one satellite area to another.

In Figure 49, Area FFF is the central area and Areas FFA and FFB are satellite areas. The RIF field for the frames going from segment 091 to segment 051 are 091/FFF-x/FFF-y/094/FFC-z/051 where x, y and z are bridge numbers of appropriate PIRs.

Topology Restrictions

PIR provides general internetworking capabilities such as:

- The PIR supports a network with a maximum of 200 LANs.
- PIR creates a flat network for level 3 addresses. Therefore, two PIR-connected LANs appear the same to an IP or IPX network or the same DECnet area. Remember this when you establish network addressing conventions.
- PIR cannot tolerate an external loop unless the loop is created with Olicom routers running PIR or unless the loop is created with the **Support Duplicate**

Physical Addresses option enabled for Source Routing bridges (**NOT** Source Routing Transparent).

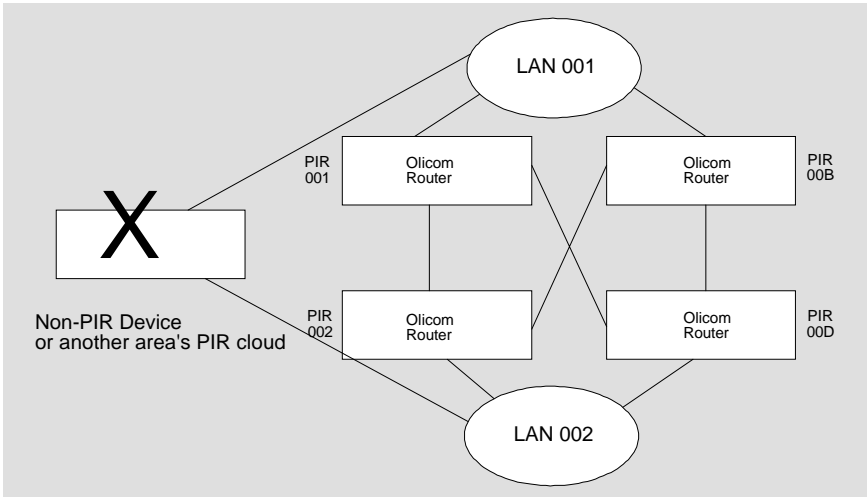


Figure 50. PIR and External Loops



9. Virtual Port

This chapter explains the role of the Virtual Port, which is a feature that allows the forwarding of traffic between a bridged network and SLCS, DLSw, IP Router, and IPX Router.

Sections

- *Overview*
- *Technical Discussion*
- *Topology Guidelines*

Overview

The virtual port (VP) allows forwarding traffic between bridged network and applications such as SLCS (XL versions 5.1 and later), and DLSw, IP Router, and IPX Router (all in XL versions 6.0 and later).

From the management point of view, the VP behaves exactly as a normal port with one exception: it is not mapped to any real XL interface.

The VP allows for simultaneous bridging and routing of IP/IPX traffic in the same XL unit (mixed routing and bridging of the same protocol). This means that use of the same protocol address (IP or IPX address) on different ports is allowed. A group of physical ports with the same IP/IPX protocol address will be seen by the IP/IPX router as if on the VP. The advantages of this include:

- With simultaneous routing and bridging of the same protocol, physical ports belonging to the VP can bridge protocol traffic between themselves and route the traffic between the VP and other ports.
- There is always one good, active IP address with which you can contact the device as an IP host. This makes it possible to maintain sessions such as SNMP/IP, DLCS, and BGP regardless of the status of any particular interface. As long as at least one interface is up and active, a session established with the VP will be maintained.

As a result, it is possible to reduce the number of IP/IPX network addresses used, which is important when the number of available IP/IPX addresses is limited or when the use of one IP/IPX network address for a remote office router (for example, bridging inside the office but routing to the central office) is possible.

From ClearSight, the IP virtual port appears as port 16.

From the console, the virtual port appears as port 15.

Technical Discussion

Simultaneous routing and bridging of IP traffic

In Figure 51, IP subnet address 128.2.19.0 / 255.255.255.0 is assigned to four LAN segments inside one remote office, with 25 host stations active on each segment. Hosts inside the remote office communicate with each other using bridging; traffic between remote offices and the central office is routed. On router R1, ports 1, 2, 3, and 4 have routing disabled on them and will be seen by routers as a single VP with one IP address and mask. Port 5 has routing enabled on it. All physical ports belonging to the VP will be invisible to the routers.

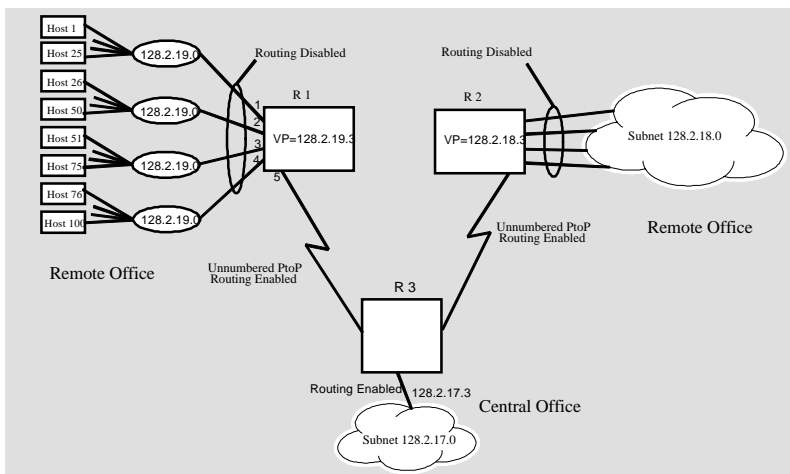


Figure 51. Simultaneous routing and bridging

IP Router acting as IP Host Station

IP router acting as an IP host station must accept and answer IP packets addressed to an IP address the host station is configured to service. This happens when you, for example, want to manage the device from a ClearSight station using SNMP/IP. Session maintenance between TELNET and BGP stations are other examples.

Prior to implementation of the VP, when you had to choose an IP address with which to communicate with a device you had to consider two cases:

- IP routing globally disabled

Global IP parameters configured on the module were used: IP global address and IP global mask to specify the address and IP network in use, and IP default gateway to support remote management (when the manager station was located in a different IP network).

- IP routing globally enabled

You had to choose one of the configured IP port addresses on a ClearSight station to address the device. Global IP parameters (address, mask, and gateway) were not available when IP routing was enabled on the module. The routing protocol or configured static routes provided connectivity.

Related implications:

- Switching from IP routing disabled on a module to IP routing enabled on that module may have required that the address used to talk to the device be changed to restore manageability (for example, when the global address did not match any of the IP port addresses).
- An inactive port could not accept and answer any traffic addressed to this port's address; the result was that if a port assigned to ClearSight/Netview/etc. went down, even though there was another physical route to the router, it could no longer be managed until you switched to the address on another port.

Example:

In Figure 52, ClearSight uses address 128.100.1.1 to monitor XL20-1; if this port goes down, ClearSight cannot maintain its session to XL20-1 over this port; you must reconfigure ClearSight and switch to another active port on XL20-1, in this case to the port with address 128.101.1.1 (see Figure 53).

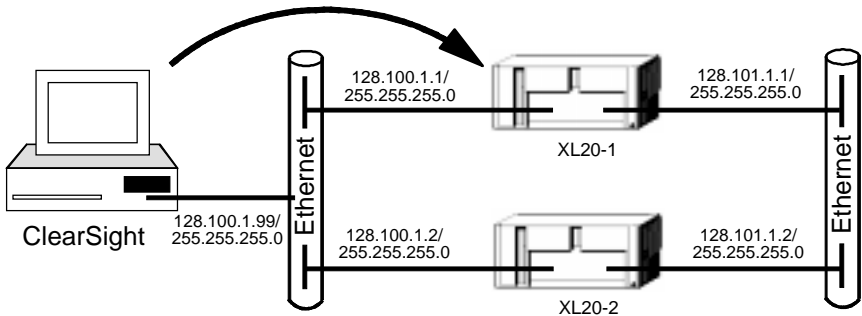


Figure 52. ClearSight monitors XL20-1 using address 128.100.1.1

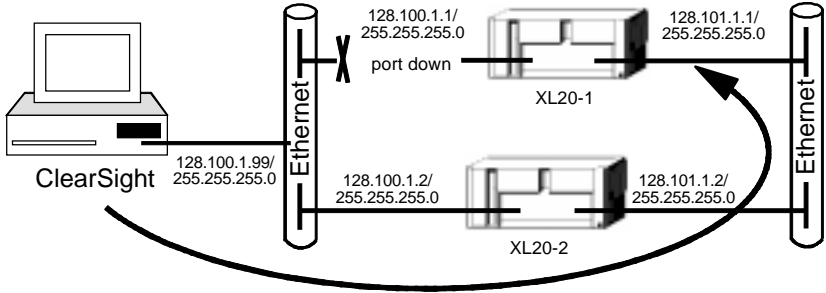


Figure 53. ClearSight monitors XL20-1 using address 128.101.1.1

Now with the VP, however, we have:

- The VP, which is always active and doesn't depend on the state of any physical port. The VP's IP address can be chosen as a host address and can provide for manageability as long as there is at least one physical route between the manager station and the managed device.
- IP router is always active and no longer can be disabled globally. IP routing can be disabled or enabled only at the port level.

Example:

ClearSight uses VP IP address 128.102.1.1 to maintain its SNMP/IP session with the device. To get through to a non-local network, the ClearSight station has to have a default gateway configured.

In Figure 54, the traffic goes through a local connection and port 1. If this port goes down, ClearSight can no longer maintain access to XL20-1 over it, but as long as the connection to the default gateway is up, the SNMP session can switch automatically to another route and remain established (Figure 55).

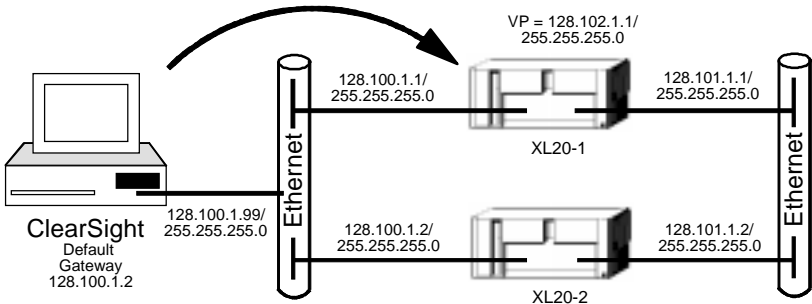


Figure 54. ClearSight monitors XL20-1 using VP's address 128.102.1.1

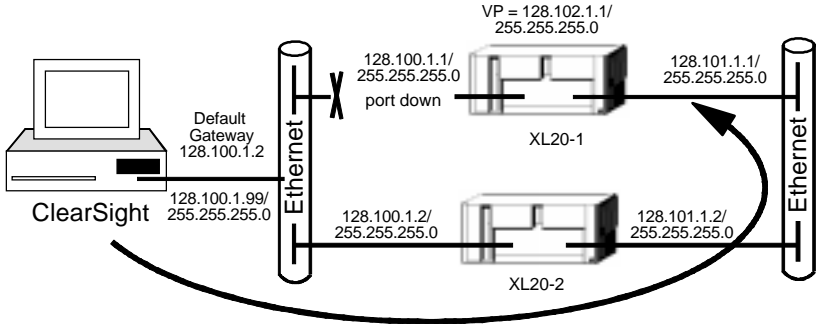


Figure 55. ClearSight monitors XL20-1 using VP's address 128.102.1.1

ICMP Ping

The VP provides one IP address active as a host address, so that there is always at least one good address to PING.

Global IP Address and Global IP Mask

With XL 6.0, the VP's IP address and mask replace the IP global address functionality, which provided IP host functionality when IP routing was globally disabled in previous XL versions. The VP is implemented as a standard IP port, and by default it is enabled for routing after HBOOT. With all other ports disabled for routing, and included into the bridge flooding procedure, the VP answers frames addressed to the module acting as IP host. The VP comes up with a default IP address of the 98 class, just as the IP global address in versions prior to 6.0 was configured after HBOOT. This address is then installed into the routing table.

Global Default Gateway

In XL 6.0, a static route to the default route replaces the IP default gateway functionality of previous XL versions. This static route to the default route's address (address 0.0.0.0, mask 0.0.0.0) is created when the default gateway address is configured. The default gateway address specifies the next hop of this static route. The default route is returned from the routing table lookup procedure when there is no information concerning the sought IP destination.

It is possible to configure more than one static route to the default route. This extends the present IP default gateway functionality. If the current default gateway disappears, any of the other configured and active routes can take over. (Note that creating a standard static route to the default route will result in the same effects as configuring the default gateway.)

Routing Table Initialization

With a virtual port, the IP routing table will always have at least following entries (even after HBOOT):

Entries	Description
0.0.0.0 / 0.0.0.0	default route; the default gateway, when configured, will appear as the default route's next hop
0.0.0.0 / 255.255.255.255	old form broadcast, discard entry
virtual port IP address and mask	local entry, with associated broadcast entries
224.0.0.5 / 255.255.255.255	OSPF multicast address, AllSPFRouters
224.0.0.6 / 255.255.255.255	OSPF multicast address, AllDRouters
224.0.0.9 / 255.255.255.255	RIP multicast address, RIP2 Routers
255.255.255.255 / 255.255.255.255	limited broadcast entry

BOOTP

When a module is configured to send BOOTP request packets after a restart, BOOTP packets are sent out on all ports disabled for routing. The possible BOOTP reply frames, the first reply received, may reset the virtual port's IP address and mask.

TCP/IP Broadcast Resolution

The router can limit the number of TCP/IP Address Resolution Protocol (ARP) broadcast messages. TCP/IP Broadcast Resolution functionality in 6.0 is restricted to have an impact only on ports where routing is disabled.

Topology Guidelines

Prior to configuring XL modules to work with simultaneous bridging and routing, the routing domain's topology restrictions must be considered. Restrictions are related to the fact that a router's behavior depends only on routing protocol algorithms, and packets are forwarded according to information kept in the router's routing table. In particular, routers do not interoperate with bridges and their spanning tree or source routing algorithms. This is important in routing domains where routing and bridging are deployed in parallel, and in redundant topologies.

Example:

In Figure 56, modules XL20-1 and XL20-2 are configured to forward both SNA and IP traffic. SNA may run either transparent or source routed bridging, IP routing is enabled on subnet 128.101.213.0 and disabled on subnet 128.101.221.0, which means that on both modules the VP addresses belong to the subnet 128.101.221.0. With these configuration requirements, forwarding of IP traffic depends very much on topology details, such as the type of bridging algorithm (transparent or source routing) deployed, spanning tree parameters on ports, spanning tree states on ports, and location of spanning tree root.

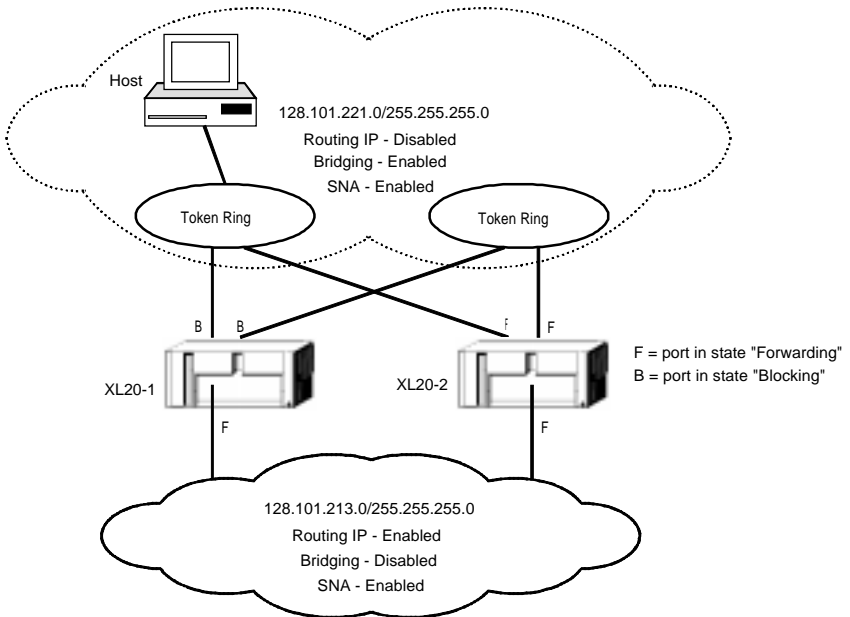


Figure 56. Routing and Bridging in parallel

In Figure 56 the worst case scenario is depicted, where IP traffic when going from routed subnet 128.101.213.0 through XL20-1 into the bridged subnet 128.101.221.0 may or may not be properly forwarded, depending on particular settings.

Note that with no bridging on routed subnet 128.101.213.0, or with no external loop connecting parts where IP routing is enabled, and with IP routing disabled (connected through one module only) IP will be forwarded properly.

Transparent Bridging

With reference to Figure 56, and with transparent bridging configured for bridged traffic, the spanning tree states on module XL20-1 ports will block the IP traffic going from routed network 128.101.213.0 through XL20-1 to IP-bridged network 128.101.221.0, which results in IP disconnectivity.

Possible manual settings

- to reconfigure routers, so as to choose module XL20-2 when routing towards bridged network,
- to affect the spanning tree states on ports

are very much topology dependent, and require thorough analysis in each case.

SR Bridging

With reference to Figure 56, and with source routed bridging configured for bridged traffic, the following are configuration restrictions for hosts located on a bridged network 128.101.221.0:

- When hosts are configured with a subnet mask (i.e., with an address belonging to network 128.101.221.0 / 255.255.255.0), hosts must direct their remote traffic to a configured default gateway. A connection to the default gateway requires address resolution using ARB type frames for ARP requests, because they are accepted on ports in state “blocking”, whereas SRB are not; once the route to the default gateway is learned, an IP session can be established - the default gateway accepts source routed, unicast packets sent to its address.
- When hosts are configured with a net mask (i.e., with an address belonging to network 128.101.0.0 / 255.255.0.0) and routers are set to act as proxy ARP servers, for the hosts to be able to communicate with XL modules they may have to generate ARB type frames when generating ARP requests to resolve addresses; SRB frames are blocked (they are neither received nor sent) on ports where spanning tree state is “blocking”; once the addresses are resolved, the source routed sessions can be established.

Simultaneous routing and bridging of IPX traffic

The Virtual Port (VP) is used by the IPX router for simultaneous bridging and routing of IPX traffic. Thanks to the VP it is easy to connect an IPX router to an existing bridged networks or to connect a bridge via WAN link with an IPX router.

Remember that in IPX router there is an IPX Internal Network which is not the same as the Virtual Port. The internal network is not linked to any IPX circuit. It is used for unnumbered point-to-point connections to properly route NetBIOS over IPX (IPX Type 20) broadcast packets. Also, the internal network must be set to run IPX Ping or IPX Diagnostic packets over Unnumbered WAN.

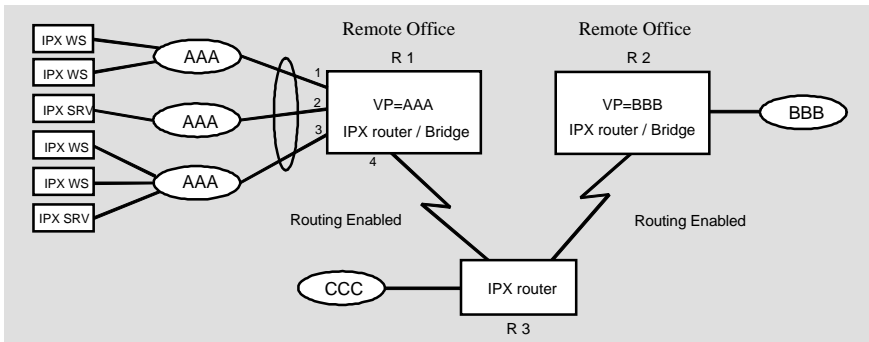


Figure 57. Sample configuration with IPX router using virtual port

On router R1 ports 1, 2, and 3 have IPX routing disabled on them and will be seen by routers as a VP with one IPX network. Port 4 has routing enabled on it. All physical ports belonging to the VP will be invisible to the IPX routers.

SLCS considerations

The virtual port (VP) is an internal XL router arrangement designed to pass LLC frames between XL bridging routines and SLCS. SLCS uses that port to receive any LLC frames from real LAN ports and to transmit LLC frames, which are subsequently directed by XL bridging routines to real LAN ports. For Ethernet frames the VP is transparent; for Token Ring frames it looks like a Token Ring segment. Therefore, in order to pass TR frames correctly, it must be assigned a unique LAN segment number.

In XL versions 5.1 and higher but prior to version 6.0, the VP was used exclusively by SLCS, it was sometimes referred to as the SLCS port, and access to it was provided in ClearSight only through SLCS windows. Starting with XL version 6.0, however, the VP is shared with other protocols and access to it has been moved to a generalized Virtual Port window accessible from the system menu. The VP's number has been fixed to 16 in ClearSight and 15 from the console across all hardware platforms.

The only parameter from the **Port Parameters** window that is important for SLCS traffic in a SR environment is the LAN Segment Number. Other parameters are irrelevant.

DLSw considerations

The virtual port (VP) is an internal XL router arrangement designed to pass LLC frames between XL bridging routines and DLSw. DLSw uses the VP to receive any LLC frames from real LAN ports and to transmit LLC frames, which are subsequently directed by XL bridging routines to real LAN ports. For Ethernet frames the VP is transparent; for Token Ring frames it looks like a Token Ring segment. Therefore, in order to pass TR frames correctly, it must be assigned a unique LAN segment number.

The only parameter from the **Port Parameters** window which is important for DLSw traffic in a SR environment is the LAN Segment Number. Other parameters are irrelevant.



10. ClearSession Protocol

This chapter explains Olicom's ClearSession Protocol and provides key information useful during configuration.

Sections

- *Overview*
- *Technical Discussion*
- *Network Considerations*

► **Note:** This manual contains reference information for Olicom's internetworking devices. For configuration and management information, refer to Olicom's ClearSight and console commands documentation.

Overview

The main idea of the ClearSession Protocol is that devices monitor one another's activity and, if a device fails, one of the others assumes its duties.

There are two fundamental objectives of ClearSession Protocol:

- guarantee no session loss when a link or an entire router or bridge fails
- with Source Route bridging, preserve traffic distribution over parallel connections when a link or bridge fails

Technical Discussion

ClearSession is supported for IP Routers and for Source Route Bridging (SRB). ClearSession configuration consists in specifying which devices within a single LAN segment should provide redundancy backup for one another. The set of such devices forms a group identified by a decimal number called *group_id*. Each device in a group has to be manually configured.

► **Note:** All ClearSession parameters are configurable through ClearSight and console commands.

ClearSession for IP Routing

ClearSession for IP Routing is designed to support IP hosts which usually do not respond well to topology changes.

In general, IP workstations communicate with remote networks (or subnets) using proxy ARP agents working on directly connected routers or using default gateway.

In a default gateway configuration, the default gateway is the router to which all remote traffic originated by the station is addressed.

In a proxy ARP environment, a workstation originates an ARP request asking for the MAC address of the destination IP address, even if the address is remote. If IP router receives an ARP request about a remote destination, it responds with its own MAC address if the destination network is known from any routing protocols.

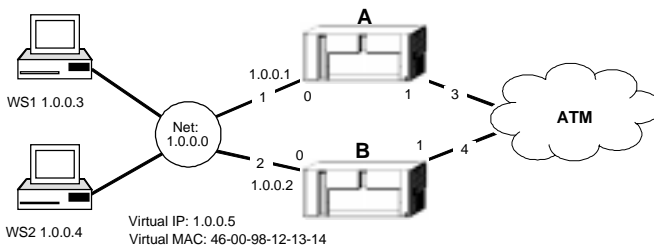


Figure 58. Sample IP Routing configuration

Figure 58 shows a sample IP Routing configuration in which workstations WS1 and WS2 communicate via routers A and B with remote networks.

Assume that workstations WS1 and WS2 communicate with remote networks via a default gateway. Both workstations must be configured manually with the IP address of the default gateway. For WS1 and WS2 this can be router A (IP address 1.0.0.1). If this router fails, both workstations cannot communicate with remote hosts until someone manually reconfigures them to use the address of router B as the default gateway (IP address 1.0.0.2).

Now assume that workstations WS1 and WS2 communicate with remote networks thanks to proxy ARP responses. To initiate a connection, workstation WS1 broadcasts an ARP request asking for the next hop of the destination IP address. It can receive an ARP reply from router A and router B. Assume it accepts a reply from router A. In this case, the entire session is routed by router A. If router A fails, the session will be broken. It can be re-established without manual reconfiguration via router B, but most workstations must be restarted manually.

ClearSession offers a solution to such problems by making it possible to maintain communication when a router becomes unavailable. It allows two or more ClearSession-configured routers to use the MAC address and IP network address of a virtual router - it does not physically exist - that represents the common target for routers that are configured to provide backup to each other.

In the configuration from Figure 58, routers A and B can be ClearSession-configured to represent a single virtual router with IP address 1.0.0.5. From among the available routers (A and B in this case), ClearSession elects an *active* router. Say it elects A. It is now router A that actually carries out the duties of the virtual router. If A fails, however, ClearSession again elects a router from those available. In this case, B remains and will be chosen.

In the first case, where workstations are configured with default gateway, default gateway should have IP address of 1.0.0.5. Initially router A can act as the virtual router and can handle remote traffic originated by stations. If A fails, router B becomes virtual router and forwards remote traffic originated by workstations. Thus existing sessions are preserved and manual reconfiguration is not required.

In the second case, where workstations communicate using proxy ARP, remote ARP requests are responded to by all routers configured to support proxy ARP and ClearSession. All responding routers specify the ClearSession group's virtual MAC address in their responses. These proxy ARP responses are not suppressed based upon ClearSession state because this could result in the lack of any proxy ARP response being generated, since these proxy ARP responses may be suppressed due to other reasons, such as split-horizon rules.

If active router A fails, router B becomes active router, forwards traffic originated by workstations, and preserves existing sessions.

ClearSession uses a priority scheme to determine which router is the default active router. To configure a router as the active router, you assign it a priority that is higher than the priority of all the other ClearSession-configured routers. The default priority is 100, so if you configure just one router to have a higher priority, that router will be the default active router.

ClearSession works by exchanging multicast messages that advertise priority among ClearSession-configured routers. When the active router fails to send a hello message within a configurable period of time, the standby router with the highest priority becomes the active router. If routers have equal priorities, the router with

the higher MAC address is selected. The transition of packet-forwarding functions between routers is completely transparent to all hosts on the network.

ClearSession can (optionally) track the availability of router ports and adjust the ClearSession priority of the router based on whether certain of the router's ports are available. When a tracked port becomes unavailable, the ClearSession priority of the router is decreased. You can use tracking to automatically reduce the likelihood that a router that has an unavailable key port will become the active router. You can also use it to resign from being the active router in case of a key port failure. In Figure 58, router A can be configured to act as an active router (priority of 105) and B as backup (priority of 100). Router A can track its port 3 and decrease its priority by 10 if this port fails. In such a configuration, in normal conditions, router A acts as active router and B as backup. If port 3 in router A becomes unavailable, its priority is decreased by 10. As a result, router B becomes active router (still priority of 100) and router A backup (priority changed to 95).

► **Note:** 'ICMP redirect' option on IP Router port should be disabled when using ClearSession IP protocol. An ICMP redirect message can redirect an IP session to run via a real router for which there is no backup, and its failure would lead to session loss.

IP-related ClearSession terms

ClearSession Protocol State

ClearSession can be in one of the following states:

- *Enabled* - the protocol is enabled on the device.
- *Disabled* - the protocol is disabled on the device.
- *Started* - the protocol is started on the device.
- *Stopped* - the protocol is stopped on the device.

ClearSession Device State

A device on which ClearSession is enabled can be in one of the following states:

- *Active* - This state means that the device works as a virtual router. This device handles virtual IP address and virtual MAC address.
- *Backup* - This state means that the device will try to become a virtual router after the current virtual router failure.

ClearSession Group ID

This number identifies a group of devices which, within a single LAN segment, will compete with one another to become a virtual router. All devices in a group monitor one another. There can be no more than 20 groups per IP device.

Virtual IP address

This is the IP address of a virtual router. The virtual router does not physically exist - instead, it represents the common target for routers that are configured to provide ClearSession backup for one another.

Priority

This parameter defines the priority for a device in a ClearSession group. The router with highest priority becomes a virtual router.

Range: 0..255. Default value: 100.

Decrement Priority

This parameter defines

- how much to decrement the router priority when a traced port becomes unavailable.
- how much to increment the router priority when a traced port becomes available.

Default value: 10.

Traced Port

This is the number of a port that will be tracked for priority calculations.

Hello Time

This is the seconds between hello messages sent by routers in one ClearSession group. Range: 0.1 second to 600.0 seconds. Default value: 1 second. If you change it, you must change it to the same value for each device within the group, and it must be less than Hold Time.

GAP Time

This is the minimum time delay between ClearSession hello periodic packets and between ClearSession hello triggered packets.

Range: 0.01..0.2 seconds. Default value: 0.1 second.

Hold Time

This is the seconds that a router waits before it declares its neighbor to be down. Range: 0.1 second to 600.0 seconds. Default value: 3.5 seconds. If you change it, you must change it to the same value in each device within the group, and it must be greater than Hello Time.

UDP Port

This is the UDP port assigned to ClearSession IP frames. Default value: 20000.

ClearSession for Source Route Bridging

ClearSession for Source Route Bridging is limited to the configurations similar to that presented on the Figure 59, the key properties of which are:

- there are two or more SR bridges (bridge 1, bridge 2, ..., bridge i) connected in parallel
- on one side they are connected to Token Ring segment (segment A), on the other - to IEEE 802.5 ELAN or ClearPath segment
- all such bridges must be configured to belong to a single group within Token Ring segment

For redundancy and load sharing reasons, bridges can be connected to more than one Token Ring segment (segment B) - additional separate group must be defined for new Token Ring segment. In such configurations, ClearSession assures that failure of a single link or entire bridge will not break active sessions established between Token Ring segments and ELAN/ClearPath segment.

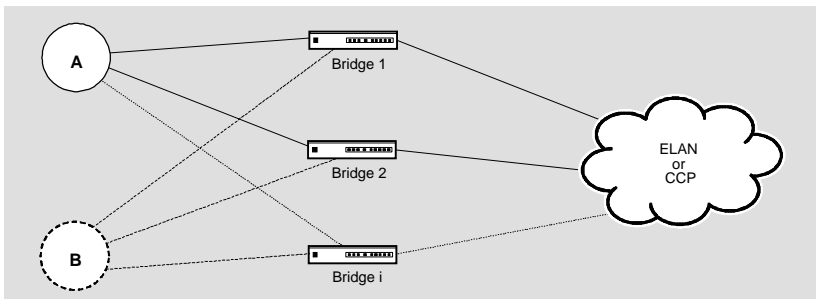


Figure 59. Sample Source Route Bridging configuration

In the situation shown on the Figure 60, the native SRB/ARB discovery protocol will equally distribute traffic between the bridges and links. The main goal of ClearSession in this case is to preserve all active sessions going through bridges 1 and 2 from segments 100 and 200 to segment 300 and vice versa even if one of the numbered

links or an entire bridge fails. To accomplish this, the time of failure detection and reaction on it must be short (a few seconds) or active sessions will be lost.

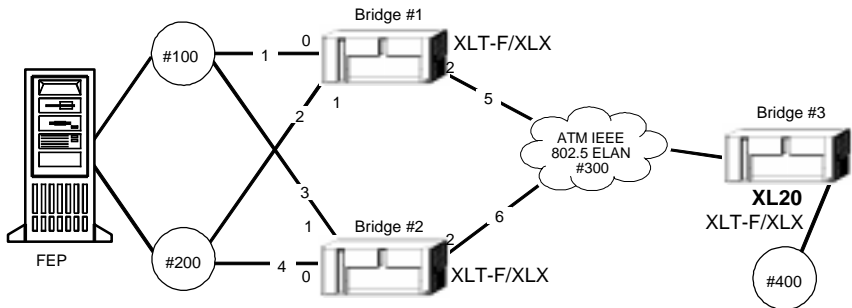


Figure 60. Sample Source Route Bridging configuration

ClearSession manages this by having bridges exchange hello messages over a configured Token Ring segment. A bridge sending a hello message indicates its configured priority and Source Route number and indicates that the bridge and its relevant interface are operational. Based on received hello messages, the bridges create a list of active neighbors on a segment, ordered by their priority. Priorities are necessary to determine the device which should perform backup actions on behalf of a failed device. If bridges have equal priorities, the bridge with the higher MAC address is selected.

In the example, assume link 1 fails. ClearSession can preserve sessions having route descriptors 100/1/300 and 300/1/100 in their paths, which would otherwise be lost. Here's how:

Bridge 1 will detect that its interface 0 is inoperational and will immediately deregister from LES Route Descriptor associated with failed port.

At the same time bridge 2 will notice absence of bridge 1 hello messages on port 1 and, after a short *hold_t* hold time interval, bridge 2 will identify itself as the only one able to act as a backup for bridge 1.

To perform its backup duties, bridge 2 will:

- update its local forwarding database to accept route descriptors 100/1/300 coming from port 1 and 300/1/100 coming from port 2
- register route descriptor 100/1 in LES to be associated with an ATM address of port 2 in bridge 2. This registration is done via an LE_REGISTER request. To sustain active sessions it is necessary to also update ARP caches of LECs participating in ELAN. This will be done using an LE_NARP_REQUEST request, which is described in the LAN Emulation Specification. This message will be broadcast by LES to all LECs participating in ELAN, and will update

their caches to associate route descriptor 100/1 with ATM address of port 2 in bridge 2

Unfortunately, according to the LAN Emulation standard, LECs are not required to react to LE_NARP_REQUEST because other known vendors do not accept LE_NARP_REQUESTs. Their devices will not work properly with Olicom's ClearSession protocol.

► **Note:** ClearSession SR can not be configured on Olicom's XLP/XLA modules because of Token Ring interface hardware limitations. These devices can accept only a single bridge number. Currently this is not a problem because these platforms do not support ATM.

SRB-related ClearSession Terms

ClearSession Group ID

This number identifies a group of devices that, within a single LAN segment, will provide redundancy backup for one another. There can be no more than 20 group definitions per bridge.

Traced port

This is a port that must be active to send hello messages. If it becomes unavailable, another bridge in the group will become the active device.

Priority

This parameter defines a priority for each devices in ClearSession group. The bridge with highest priority becomes an active device. If devices have equal priorities, the device with the higher MAC address is selected.

Range: 0 to 255. Default: 100.

Hello Time

This is an interval in seconds between hello messages sending by devices in one ClearSession group. Range: 0.1 second to 600.0 seconds. Default: 0.4 seconds. If you change it, you must change it to the same value in each device within the group, and it must be less than Hold Time.

GAP Time

This is the minimum time delay between Clear Session hello periodic packets and between Clear Session hello triggered packets.

Range: 0.01 to 0.2 seconds. Default value: 0.1 second.

Hold Time

This is a duration in seconds that a device waits before it declares the neighbor to be down. Range: 0.1 second to 600.0 seconds. Default: 3.5 seconds. If you change it, you must change it to the same value in each device within the group, and it must be greater than Hello Time.

Network Considerations

Consider the network diagram shown in the Figure 60. Bridge 1 and bridge 2 are configured as ClearSession devices and bridge 1 is the active device. When bridge 1 fails, bridge 2 starts to back up bridge 1. To update LEC caches, it generates an LE_NARP_REQUEST to inform listeners about the new owner of bridge 1's route descriptor. It is able to send a properly built LE_NARP_REQUEST since it knows (via hello message exchange) an ATM address of bridge1's LEC ATM address. If at the same time bridge 2 fails and bridge 1 comes up, it is impossible for bridge 1 to know that the previous owner of its own Route Descriptor was bridge 2. Therefore it can not send proper a LE_NARP_REQUEST.

You can use Olicom TLV, which informs whether it is legal to send an LE_NARP_REQUEST with a NULL TARGET_ATM_ADDRESS field (TARGET_LESS_NARP). If the LEC is populated with Olicom devices, then the presence of this TLV will inform ClearSession that it is legal to generate a LE_NARP_REQUEST without a TARGET_ATM_ADDRESS field and that LECs will accept such requests. When there are devices from other vendors connected to the ELAN that could misunderstand such an LE_NARP_REQUEST, absence of Olicom's TLV will inform LECs to fully conform to the LANE specification and ClearSession not to send an LE_NARP_REQUEST.



11. Introduction to SNA and NetBIOS

This chapter provides a general introduction to the two features that allow Olicom routers to participate in SNA and NetBIOS networks. For more information on these features, see the appropriately named chapters.

Sections

- *Overview*
- *SDLC/HDLC PassThrough*
- *SLCS*
- *DLSw*

Overview

Olicom routers support several features through which they participate in SNA or NetBIOS networks.

- SDLC/HDLC PassThrough allows routers and their WAN links to carry SNA traffic, substituting for SNA WAN links.
- SLCS (SNA Link Conversion Services) converts between SDLC and LLC2 data link layer protocols and frame formats.
- DLSw (Data Link Switching) - a method for handling SNA and NetBIOS data traffic.

While this book explains and makes reference to some terms, concepts, and products, it does not provide enough information for an SNA or NetBIOS novice to successfully configure a complex network.

SDLC/HDLC PassThrough

SDLC/HDLC PassThrough allows routers to multiplex SNA (SDLC or HDLC) traffic over WAN links with LAN traffic. Routers at both ends of the link:

- Encapsulate SNA frames inside token ring frames before sending them over the WAN link
- Deencapsulate token ring frames (when appropriate) before sending the frames to an SDLC or HDLC device.

These features allow routers and their WAN links to replace SNA WAN links.

PassThrough has two modes of operation:

- *General* - Supports HDLC traffic in point-to-point WAN topologies
- *SDLC* - Supports SDLC traffic in point-to-point, virtual multidrop or local multidrop topologies.

See chapter 12, *SDLC/HDLC PassThrough* in this volume for more information.

SLCS

SLCS allows a router to connect an SDLC data stream on one port to an LLC2 data stream on another port.

Because SDLC and LLC2 have different addressing methods, SLCS provides session definition tools through which LLC2 addresses (MAC addresses) are mapped to SDLC addresses.

SDLC requires that each end of a session be either a *primary* node or a *secondary* node. SLCS enables routers to take either role.

There are many network topologies and SNA devices supported by SLCS. Complete configuration information for the following PU 2.0 configurations (Figure 61) is provided in chapter 13, *SLCS* in this volume.

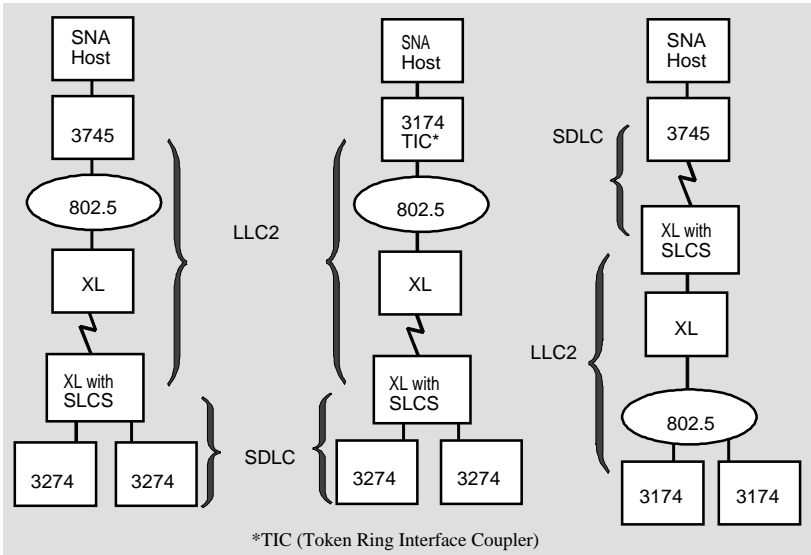


Figure 61. Three Typical SLCS configurations

SLCS also supports configurations involving PU 2.1 and PU 1.0 devices. For more information, see chapter 13, *SLCS* in this volume.

DLSw

Olicom's implementation of DLSw is based on RFC 1795 and RFC 1434 standards.

DLSw makes two communicating end stations each appear adjacent to the other on a shared data link. The data link can be one of the following: LLC type-1 or LLC type-2 (on Token Ring and Ethernet), or SDLC. DLSw combines two data links by terminating each logically and relaying the data between them using TCP. A DLSw router appears to its local end stations as a collection of remote end stations.

DLSw uses switch-to-switch protocol (SSP) to transfer data between partner routers. The frame type sent by the end stations can be either an LLC (Ethernet or Token Ring) or SDLC frame. When such a frame from an origin station arrives at a DLSw router it is converted into the SSP frame format. The target DLSw makes the opposite transformation and sends it to the target end station as an LLC or SDLC frame (depending on the target station).



12. SDLC/HDLC PassThrough

This chapter explains Olicom's SDLC/HDLC PassThrough.

Sections

- *Overview*
- *Technical Discussion*
- *Network Considerations*

Overview

Olicom routers support SDLC/HDLC PassThrough. PassThrough allows central host-based communications to share the same WAN interface as internetworked LAN traffic.

Olicom's PassThrough supports the following:

- ***General or HDLC PassThrough mode*** -- Router supports point-to-point HDLC or SDLC traffic.
- ***SDLC-Specific mode*** -- Router supports both point-to-point and multipoint SDLC applications.

Technical Discussion

This section discusses Olicom router support of the following PassThrough modes:

- General PassThrough mode
- SDLC-Specific PassThrough mode

General PassThrough Mode

General PassThrough mode can be implemented in any point-to-point SDLC or HDLC network. This mode does not use addresses or data within the SDLC/HDLC frame. General PassThrough mode is manipulated through ClearSight where a circuit number and a station number must be assigned to each interface. The circuit number must be the same for both sides of the connection (i.e., both interfaces). The station number must have the value 1 for one of the interfaces and 2 for the other.

When the Olicom router receives a frame on a PassThrough interface it encapsulates it into a token ring LAN packet (Figure 62). The source address is created by the Olicom router based on the circuit number and the station number. Because it is a point-to-point connection, the destination address is assumed to be Station 2 if the packet is received on Station 1 and vice versa. Once the LAN packet is created, the Olicom router treats it as any other LAN packet. To the two communicating end-stations, this routing process is transparent.

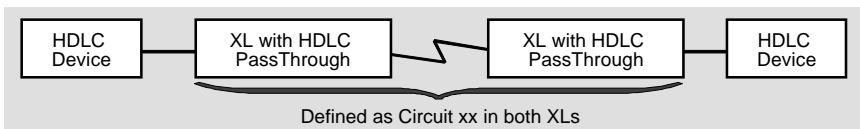


Figure 62. General PassThrough

SDLC-Specific PassThrough Mode

In SDLC-Specific mode, the SDLC station address contained in the SDLC frame is used for routing the packet. This mode supports both point-to-point and multipoint SDLC applications. As with General PassThrough mode, each port enabled for a particular SDLC connection must have the same PassThrough circuit number. The station number assigned to each port must be the SDLC station number assigned to the downstream device (polling address), e.g., C1 (hex). The station number for the connection to the FEP must be zero.

After the circuit numbers and stations are defined, SDLC-Specific PassThrough Mode operates much like General Mode. The source address is created from the circuit number and the station number. If the source station is Station 0 (the FEP),

then the destination station is extracted from the SDLC packet. If the source station is not 0 (the FEP), then the destination station is zero.

Figure 63 shows a Point-to-Point Passthrough connection in SDLC-Specific mode. Figure 64 shows a multidrop PassThrough connection in SDLC-Specific mode.

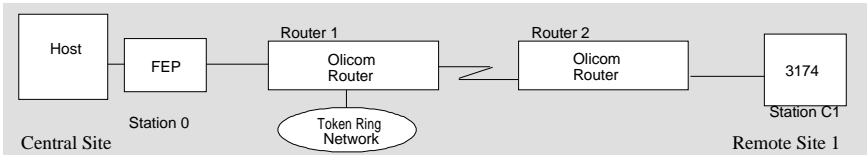


Figure 63. Point-to-Point in SDLC-Specific Mode

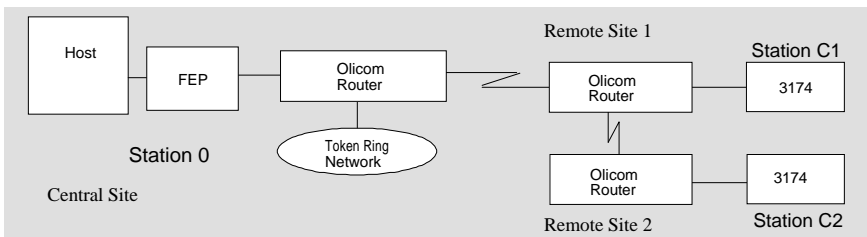


Figure 64. Multidrop PassThrough in SDLC-Specific Mode

Network Considerations

This section contains examples of the benefits of using PassThrough mode, as well as some design considerations.

Configuration Examples

PassThrough allows central host-based communications to share the same WAN interface as internetworked LAN traffic. This allows you to replace host-based communications paths with local connections to an Olicom Router. For example, Figure 65 shows a typical SDLC configuration with a 3174 cluster controller connected to a host using a Front End Processor (FEP). Figure 66 shows a token ring LAN configuration.

Using SDLC/HDLC PassThrough, you can replace host-based communications paths with local connections to the Olicom Router. Then, SDLC/HDLC traffic is multiplexed over the WAN link along with other LAN traffic as shown in Figure 67.

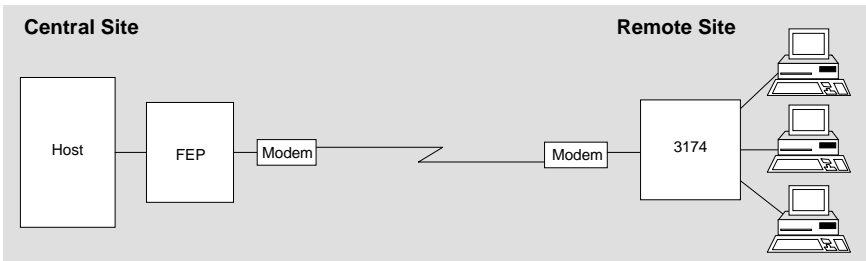


Figure 65. SNA Network

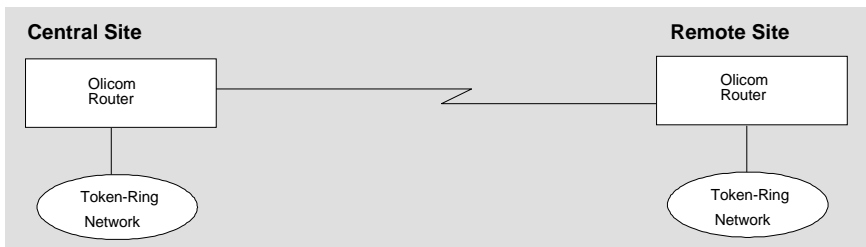


Figure 66. Olicom LAN Multiport Network

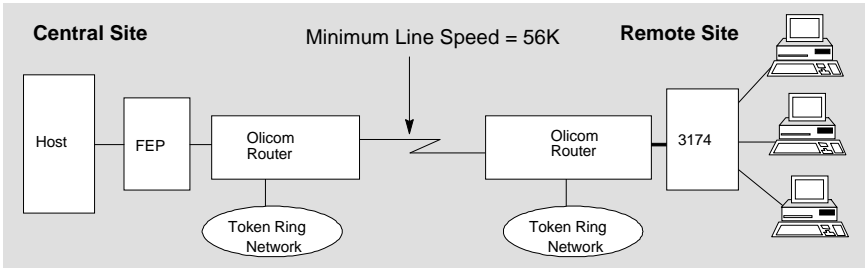


Figure 67. Combined Networks Using PassThrough for SNA Traffic

Design Considerations

Using PassThrough, there are four basic network designs that you can use:

- General Point-to-Point
- SDLC Point to Point
- SDLC Virtual Multidrop
- Local Multidrop

General Point-to-Point

This mode supports any HDLC/SDLC data stream between two points in the network (Figure 68).

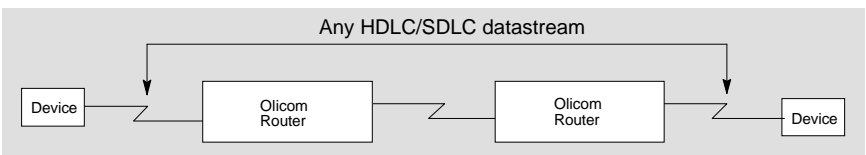


Figure 68. General Point-to-Point Mode

SDLC Point-to-Point

This mode supports any SDLC data stream between two points in the network (Figure 68).

SDLC Virtual Multidrop

This mode supports a number of logically multidropped SDLC stations (secondary) connected to a single SDLC master (primary). Each station connects to a different port on an Olicom router usually at different physical locations.

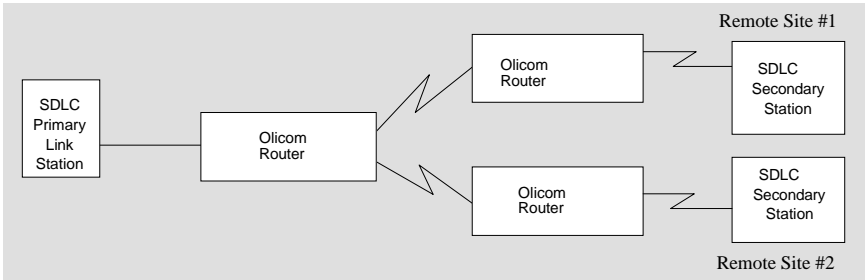


Figure 69. SDLC Virtual Multidrop Mode

Local Multidrop

Using General PassThrough mode, multiple SDLC stations can connect to a primary SDLC station using a modem sharing device.

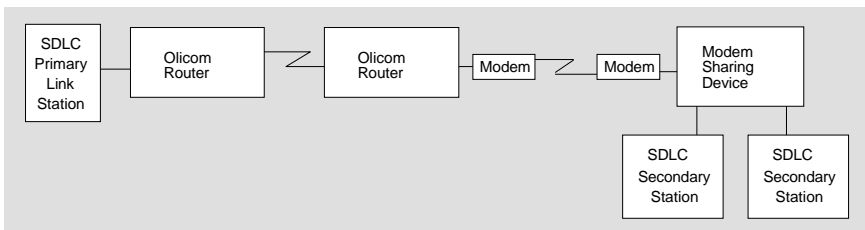


Figure 70. Local Multidrop Mode

Topology Restrictions

Local multidrop configurations can only be implemented in General PassThrough mode. However, virtual multidrop can only be implemented in SDLC-Specific mode. For more information, refer to *Technical Discussion* on page 146.



13. SLCS

This chapter provides reference material that explains Olicom's SLCS (SNA Link Conversion Services) and provides key information useful during implementation and configuration.

Sections

- *Overview*
- *Typical SLCS Configurations with PU 2.0 Devices*
- *Timeout Avoidance through Local Termination*
- *PU 2.1 and PU 1.0 Devices*
- *LLC and SDLC*
- *Connecting SLCS to Ports, Interfaces, and the Packet Switch Engine*
- *SLCS and Bridging*
- *Two Sides of a SLCS Session*
- *Host, FEP, SLCS and Controller Parameters for PU 2.0 Devices*
- *FID2 Segmentation, Multidrop and Group Polling*
- *XID Identifiers During SLCS Session Establishment*
- *NetView Support*
- *Sample Complex Topology*

Overview

SLCS software allows XL and ILAN routers to link SNA SDLC (Synchronous Data Link Control) sessions with LLC type 2 (LLC2) sessions. SLCS make these links by:

- Converting between SDLC and LLC2 frame formats
- Connecting PU type 2.0 and 2.1 devices
- Bridging (or routing using PIR) packets between XL SLCS routers

SLCS also allows SNA PU type 1.0 nodes using SDLC to communicate across an LLC2 network segment.

Table 12 provides a summary of additional SLCS features.

SLCS Feature	Description
Local Termination	Termination of SDLC and LLC2 sessions locally to avoid exceeding timeout parameters.
ClearSight Configuration	All SLCS configuration and monitoring is handled by Olicom's SNMP-based ClearSight.
Bridge Modes	Full support of IBM source routing, transparent spanning tree bridging and source routing transparent.
LLC2 over the links:	LLC2 is supported over Token Ring, Ethernet/802.3, Frame Relay, X.25, ATM, CC Point-to-Point, ISDN.
SNMP	Support of SNMP through an Olicom private MIB.
Segmentation of FID2 frames	Segmentation of FID2 frames (either LLC2 or SDLC) when necessary.
Error Checking	Full support for all error recovery procedures required by the two protocols.
Efficient Polling	Support for multidrop, group polling, and poll spoofing.
SDLC Multidrop	Support for multiple stations on an SDLC connection.
Flow Control	Support for RNR (Receive Not Ready) protocols to prevent transmitter from overrunning receiver.
NetView Agent	SLCS's NetView agent alerts the SNA host of unusual events during conversions and accepts NetView operator commands for defined SLCS sessions.

Table 12. Additional SLCS Features

Typical SLCS Configurations with PU 2.0 Devices

SLCS is typically used to connect PU 2.0 devices, including hosts, FEPs (Front End Processors) and many cluster controllers.

- **Note:** SLCS also supports PU 2.1 devices and a special configuration using PU 1.0 devices; see *PU 2.1 and PU 1.0 Devices* on page 158 later in this chapter for details regarding these PU types.

LLC2-to-SDLC

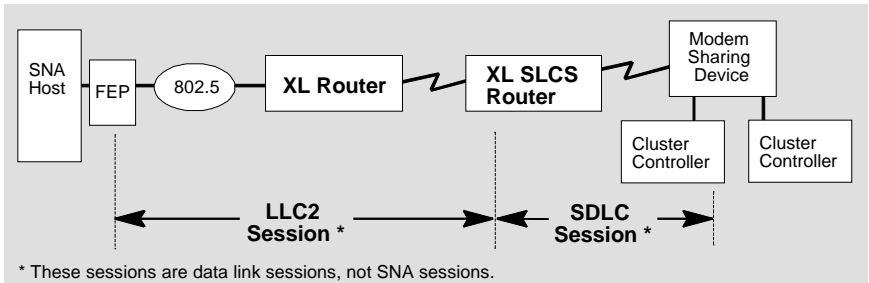


Figure 71. SLCS LLC2-to-SDLC Session

The typical SLCS implementation (Figure 71) above involves:

- A Token-Ring attached FEP (Front End Processor) uses LLC2 to communicate across a Token Ring network to the Olicom XL SLCS router. (The packets are bridged from the XL router to the XL SLCS router.)
- The XL SLCS router converts the packets into SDLC format and sends them over the serial WAN link to the modem sharing device and on to the cluster controller.

As in Figure 72, the FEP *thinks* it is communicating with downstream LLC2 cluster controllers and the downstream cluster controllers *think* they are communicating with a primary SDLC node (that is, a FEP). The ability of the SLCS process to masquerade as other devices is known as spoofing.

This configuration shows a single FEP having SLCS sessions with two cluster controllers, although only one SDLC session is indicated. This is an example of a multidrop configuration. (See *FID2 Segmentation, Multidrop and Group Polling* on page 176)

SDLC-to-LLC2

The typical SLCS SDLC-to-LLC2 implementation pictured in Figure 72 involves:

- A FEP uses SDLC to communicate to an Olicom SLCS router.
- The SLCS router converts the SDLC packets into LLC2 packets and bridges them to the XL router.
- The XL router bridges the LLC2 packets onto the 802.5 LAN, where the cluster controller picks them up.

Again, the SLCS process is spoofing the two end-stations that have the overall session. The FEP *thinks* it is communicating with an *SDLC* cluster controller and the cluster controller *thinks* it is communicating with an *LLC2* FEP (that is, a FEP that is attached to a Token Ring through a TIC (Token Ring Interface Coupler)).

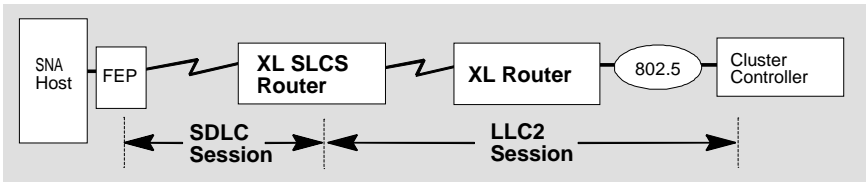


Figure 72. SLCS SDLC-to-LLC2 Session

SLCS and non-SLCS Sessions

Figure 73 shows a mix of SLCS and non-SLCS sessions. The XL SLCS router handles both SLCS sessions and non-SLCS sessions while the XL router does not use SLCS at all.

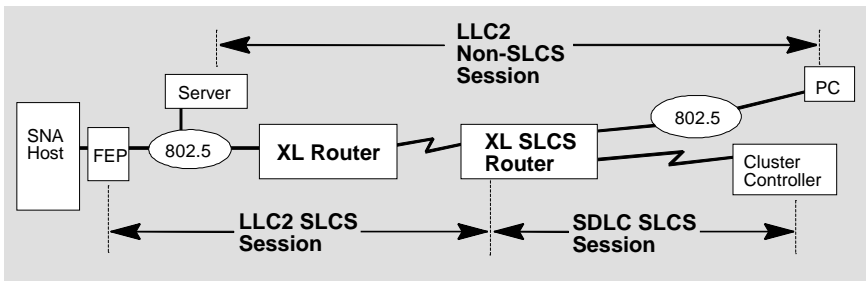


Figure 73. SLCS Configuration with SLCS and Non-SLCS Sessions

Timeout Avoidance through Local Termination

SLCS also supports local termination of SDLC and LLC2 sessions. Local termination helps avoid session disconnection caused by exceeded timeout limits. These timeout problems sometimes occur when network congestion or slow WAN links delay SNA packets longer than allowed by certain SNA nodes, such as FEPs and cluster controllers.

SLCS local termination avoids such timeouts by:

- Converting an SDLC session into an LLC2 session at one router and then converting the LLC2 session back into an SDLC session at the next router. (Shown in Figure 74.)
- Or, when SDLC is not used: terminating one LLC2 session at the SLCS process and linking it with a second LLC2 session that completes the link. (Shown in Figure 75.)

SDLC-LLC2-SDLC

Figure 74 shows one configuration in which SLCS prevents timeouts through local termination of SDLC sessions.

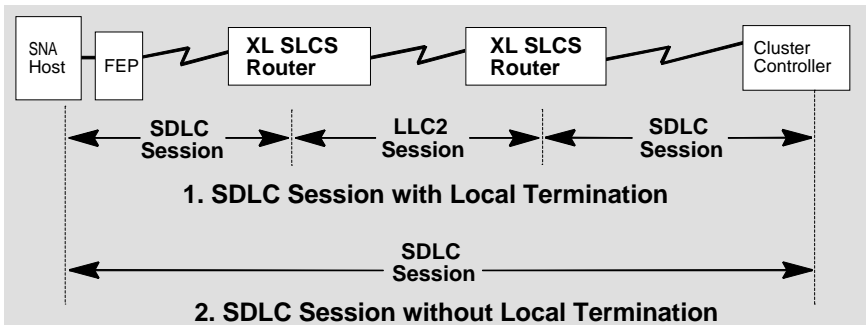


Figure 74. SLCS-based Local Termination

The diagram on Figure 74 shows two configurations: “SDLC Session *with* Local Termination,” and “SDLC Session *without* Local Termination.”

1. SDLC Session *with* Local Termination shows each router terminating two SDLC links using SLCS. This assures timeout limits are not exceeded. The following shows the sequence of events and clarifies how local termination prevents timeouts:
 - The FEP initiates an SDLC link and the XL SLCS router terminates the link.
 - The XL SLCS router converts packets into LLC2 format and bridges them to the second XL SLCS router.

- The second XL SLCS router converts the packets back into SDLC format and initiates a second SDLC session across a WAN link to the final cluster controller.
2. SDLC Session without Local Termination shows a scenario in which Olicom's SDLC PassThrough feature is enabled. SLCS is not used in this configuration. The following shows the sequence of events and explains how timeouts are possible:
- The FEP initiates a single SDLC link with the cluster controller.
 - This link is made possible by the SDLC PassThrough feature. However, a slow WAN link might cause enough delay that the SDLC session is disconnected by the FEP because its timeout period has elapsed before it has received a response packet.

LLC2-to-LLC2

Figure 75 shows a configuration in which SLCS can prevent timeouts through local termination of LLC2 sessions.

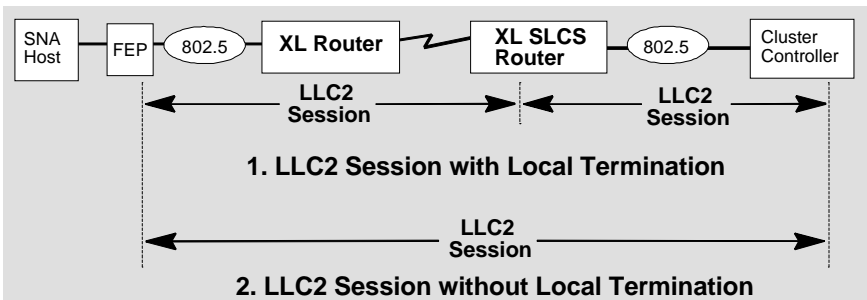


Figure 75. LLC2-based Local Termination

The diagram on Figure 75 shows two configurations: “LLC2 Session *with* Local Termination,” and “LLC2 Session *without* Local Termination.”

1. LLC2 Session *with* Local Termination shows the SLCS router terminating an LLC2 link. This assures timeout limits are not exceeded. The following shows the sequence of events and clarifies how local termination prevents timeouts:
 - The FEP initiates an LLC2 session across the 802.5 LAN and the session is terminated by the XL SLCS router. This portion of the network includes a WAN link that may be slow and could cause timeouts. Because timeout parameters can be set for FEPs (but may not be for cluster controllers), the LLC2 session does not need to be terminated prior to the WAN link.

- The XL SLCS router initiates an LLC2 session with a cluster controller attached to a local 802.5 LAN. Timeout parameters generally cannot be set for cluster controllers. Terminating the link at the local SLCS router prevents the slow WAN link from causing the cluster controller's timeout period to be exceeded.
2. LLC2 Session *without* Local Termination shows a configuration in which LLC2 sessions are not terminated and exceeding timeout limits becomes more likely. SLCS is not used in this configuration.
- The FEP initiates an LLC2 link with the cluster controller. This link is made possible because each XL simply bridges the packets as LLC2 packets.
 - Exceeding a timeout limit is possible if any WAN link is slow enough (or if it is so congested) that the FEP does not receive a response packet before a timeout limit is exceeded.

The leftmost XL router in Figure 75 could also run SLCS and could terminate the LLC2 session between the FEP and the rightmost XL. In this case there would be three LLC2 sessions: one from the FEP to the leftmost router, one from the leftmost router to the rightmost router, and one from the rightmost router to the cluster controller.

PU 2.1 and PU 1.0 Devices

Every device in an SNA network must be a defined PU type. Commonly used PU types include PU 2.0 and PU 2.1. SLCS now provides the ability to link PU 2.1 devices across mixed networks of SDLC and LLC2 network segments. Support for PU 2.1 extends SLCS to include the mid-sized IBM AS/400, to its associated 5494 and 5394 controllers, and to PCs running OS/2 with Communications Manager or PC Support/Windows.

PU 2.1

PU 2.1 systems are elements of IBM's peer-to-peer networking plan. This plan has three key elements: APPC (Advanced Peer-to-Peer Communications), APPN (Advanced Peer-to-Peer Networking) and PU 2.1 itself. The relationship between these elements is as shown in Figure 76.

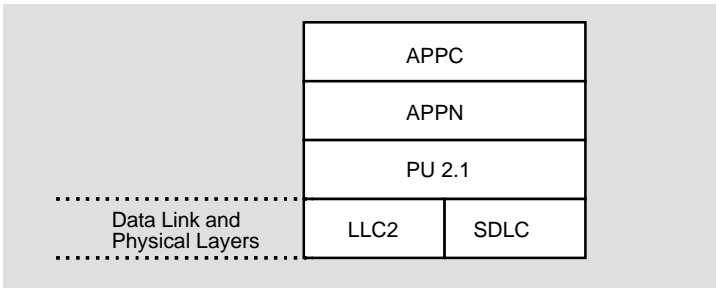


Figure 76. PU 2.1 Protocol Stacks

Figure 76 shows that PU 2.1 devices participate in the APPC/APPN peer-to-peer networking plan. PU 2.1 devices can run over the LLC2 or the SDLC lower layers (the physical and data link layers). SLCS routers can be used to connect PU 2.1 systems using LLC2 and those using SDLC, as shown in Figure 77.

PU 2.1 with SLCS

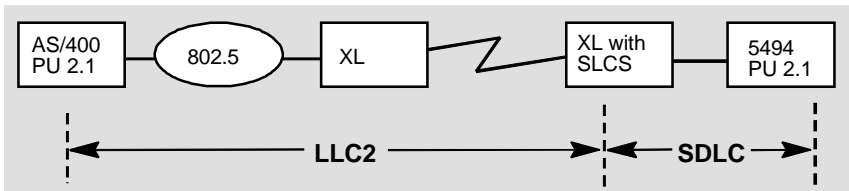


Figure 77. Typical SLCS Configuration with PU 2.1 devices

Figure 77 shows a typical configuration in which PU 2.1 devices are connected using SLCS. Both the AS/400 system and the 5494 cluster controller are behaving

as PU 2.1 devices. This configuration is useful when there are existing 5494 cluster controllers that only support SDLC and that must communicate with LLC2-attached AS/400 systems.

Note that SLCS routers may be linked in series allowing connections as follows: LLC2-to-LLC2-to-LLC2; or, SDLC-to-LLC2-to-SDLC. These are examples of local termination (See the section: A Special Configuration: *Timeout Avoidance through Local Termination* on page 155.)

PU 1.0

Among the PU types defined in SNA is the older PU type 1.0. PU 1.0 devices cannot use the LLC2 data link layer protocol, and therefore their traffic may not travel natively across links that use LLC2, such as Token Ring and Ethernet/802.3 LANs. 5294 and 5394 cluster controllers function as PU 1.0 devices when attached to AS/400 systems.

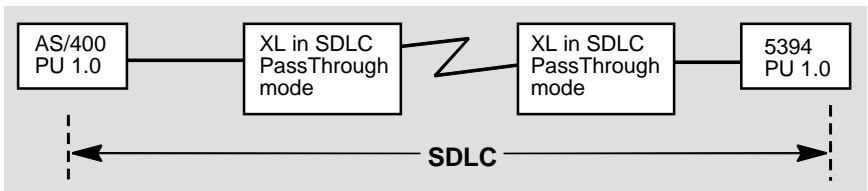


Figure 78. PU 1.0 Devices without SLCS

Figure 78 shows an SDLC session between PU 1.0 devices *without* SLCS. This configuration can result in SDLC session disconnection if the session timers are exceeded. To overcome this problem, you can use SLCS to locally terminate the SDLC sessions and thereby prevent session timeouts, as shown in Figure 79.

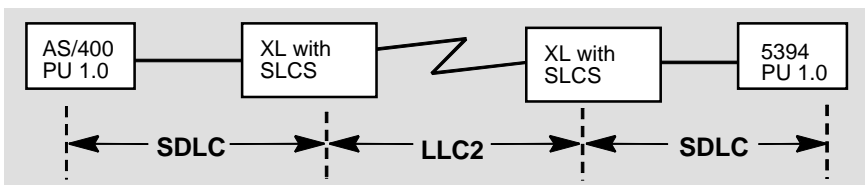


Figure 79. PU 1.0 devices with SLCS

LLC and SDLC

SLCS is essentially a gateway between two differing data link layer protocols: SDLC and LLC2. There are two versions of LLC: LLC1 and LLC2. The next subsection compares LLC1 to LLC2. The following subsection compares LLC2 to SDLC.

LLC1 and LLC2

LLC is one portion of the data-link layer protocol established by IEEE for use over 802 LANs. (For more information on LLC and the data link layer, see chapter 1, *Internetworking Principles* in volume 1.) Various upper layer protocols (that is, network layer and higher) use different 802.2 LLC sublayer protocols, for instance:

- IP and IPX use LLC1
- SNA and NetBIOS use LLC2

The table below illustrates the key differences between LLC1 and LLC2.

Feature	LLC1	LLC2
Used over 802 LANs	Yes	Yes
Peer-to-peer	Yes	Yes
Connection-oriented	No	Yes
Frame acknowledgments supported	No	Yes
Provides sequence control for packets	No	Yes
Provides error-free connection	No	Yes

Table 13. Key differences between LLC1 and LLC2

LLC2 shares more qualities with SDLC than does LLC1. LLC2 and SDLC are connection-oriented, provide sequence control, and provide error-free connections. These shared characteristics make LLC2 a good fit with the SNA environment. Thus IBM decided to use LLC2 when sending SNA packets over 802 LANs.

LLC2 and SDLC

While IBM uses LLC2 for SNA traffic over 802 LANs, the firm selected SDLC for SNA traffic over serial WAN links. Both SDLC and LLC2 are based on the HDLC (High-level Data Link Control) standard. However, LLC2 uses the Asynchronous Balanced Mode (ABM) of transmission, while SDLC uses the Normal Response Mode (NRM). Table 14 provides further comparison of SDLC and LLC2.

Feature	SDLC	LLC2
Peer-to-peer	No	Yes
Connection-oriented	Yes	Yes
Acknowledgments supported	Yes	Yes
Configurable window sizes	Yes	Yes
Provides sequence control for packets	Yes	Yes
Provides error-free connections	Yes	Yes
Transmission type	NRM	ABM

Table 14. Further comparison of SDLC and LLC2

By these criteria the two protocols are very similar, but one key difference is that SDLC is hierarchical in nature, whereas LLC2 is peer-to-peer. The hierarchical nature of SDLC is evident in the division of SDLC sessions into two nodes, a *primary* node and a *secondary* node. Only primary nodes can start or end an SDLC session. XL routers supporting SLCS are able to function as both primary and secondary SDLC nodes.

SLCS links are defined using ClearSight. While individual sessions are transient, the logical links remain in place until they are changed in ClearSight. (See SLCS and Bridging)

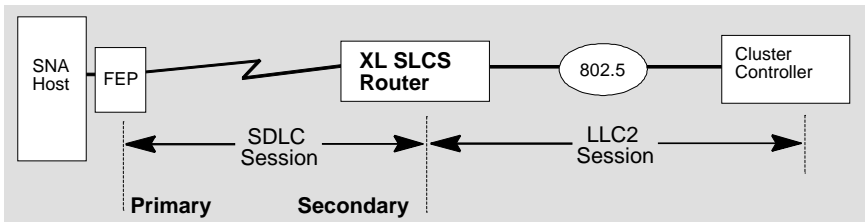


Figure 80. Primary and Secondary SDLC Nodes

Figure 80 shows the SNA relationships of the nodes at both ends of the SDLC session. The FEP is a primary node. As such, it initiates the SDLC session. The SLCS software in the router responds to the primary nodes and establishes a session with the router functioning as a secondary node. When packets are received, SLCS converts them into LLC2 format. The router then bridges the packets using the appropriate bridging mode (transparent or source routing).

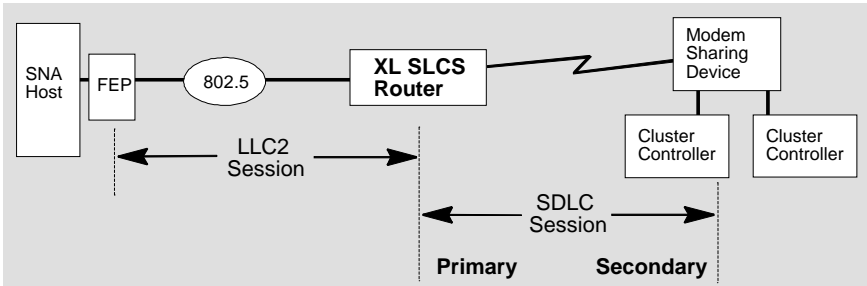


Figure 81. Primary and Secondary SDLC Nodes

Figure 81 shows the SNA relationships of the nodes at both ends of an SDLC link. In this case, the SDLC session is between a router and a cluster controller. When the SLCS router receives bridged LLC2 packets from the FEP, it initiates an SDLC session as a primary node. The downstream cluster controller responds as a secondary node.

When packets are sent from a SLCS session in one router to a SLCS session in another:

- The data link protocol is LLC2.
- The packets are bridged using whichever bridging mode (transparent or source routing) is appropriate, or they are routed using PIR.

Any traffic not originating in SLCS sessions, for instance the traffic moving between the PC and the Server in Figure 83, can be routed or bridged between XL routers.

SLCS Frame Conversion

Because both LLC2 and SDLC are data link layer protocols, a conversion between them is relatively straightforward. SLCS simply strips the header and trailer of one data link layer off each frame and replaces them with the header and trailer of the other data link layer, as shown in Figure 82, when converting SDLC into LLC2.

When necessary, large frames are also segmented into smaller frames. (See *FID2 Segmentation, Multidrop and Group Polling* on page 176.)

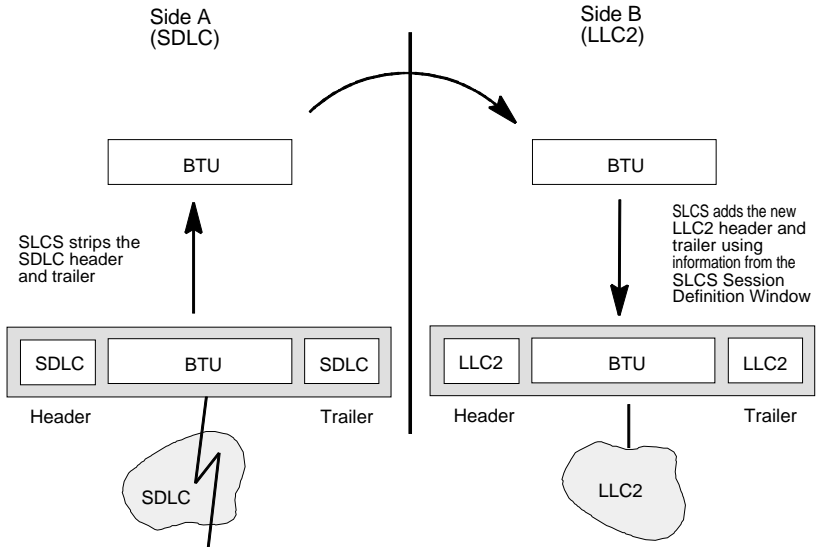


Figure 82. Converting a Frame from SDLC to LLC

Connecting SLCS to Ports, Interfaces, and the Packet Switch Engine

SLCS is a software process that fits in the router with many other processes. It is connected to physical interfaces as well as to ports that lead to other processes in the router. In order for SLCS to operate, the SLCS Conversion Engine must be connected to a WAN port and to the Packet Switch Engine.

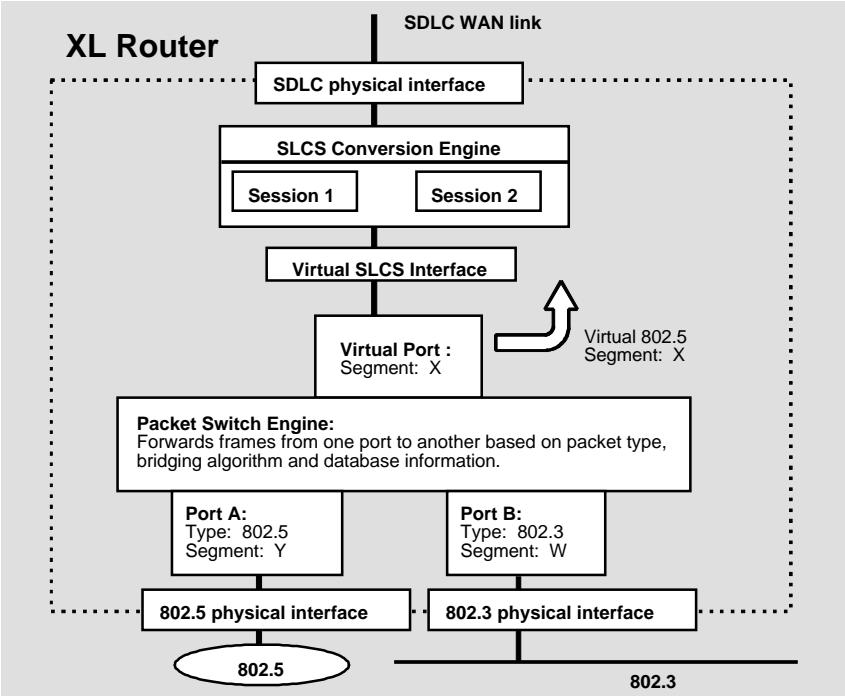


Figure 83. Connecting SLCS to Ports, Interfaces and Processes

Figure 83 shows the connections between SLCS, ports, interfaces and the Packet Switch Engine. (The segment numbers in Figure 83 are shown as letter variables. You must set these to actual values.)

On one side SLCS is connected to an SDLC physical interface, while on the other it is connected to the Virtual Port, which always is (has) number 16(ClearSight). (ILAN routers use port 8 as the SLCS default port.)

All traffic sent from the Packet Switch Engine to the SLCS Conversion Engine, or from the SLCS Conversion Engine to the Packet Switch Engine, travels through the Virtual Port.

SLCS and Bridging

As previously noted, SLCS always involves either bridging or routing with PIR.

The Packet Switch Engine determines each packet's type (Ethernet/802.3 or 802.5) and switches it from one port to another depending on:

- The bridging algorithm the packet is using
- The packet's addressing and bridging data
- Any additional information the switch provides from its database

Each port used by the Packet Switch Engine has a segment number associated with it. This segment number is used when forwarding source routed packets, but not when forwarding transparently bridged packets (which do not use segment numbers). You set these segment numbers, including the segment number for the Virtual Port, using ClearSight.

If the algorithm is:

- *Transparent*: The packet is switched to the port that is associated with the destination MAC address. When the packet is transparently bridged to (and across) SLCS, the destination address is the Local MAC address listed in the LLC2 side of the SLCS Session Definition screen. Thus, the Packet Switch Engine uses the Virtual Port for any packets transparently bridged across SLCS.
- *Source routing*: The packet is switched to the port associated with the next segment number listed in the packet's RIF field. When the packet is source routed across SLCS, that segment number is the Virtual Port's segment number.
- *Source routing transparent*: If the packet is a source routed packet, it is switched as a source routed packet; if it's transparently bridged, the packet is switched using the transparent bridging mode. Note that when Heterogeneous Bridging is used, each Ethernet/802.3 segment or WAN segment must have a segment number set for it to allow it to participate in the source routing scheme. Thus, in Figure 83, the port leading to the 802.3 LAN must have a segment number set for it.
- *PIR*: The frame is stripped of the PIR encapsulation information and forwarded to the Virtual Port for passage to SLCS. If the router is in PIR/transparent mode, the switching decision is based on the destination MAC address. If the router is in PIR/source routing mode, the switching decision is based on the segment number in the RIF field.

Two Sides of a SLCS Session

SLCS connects two links, usually an SDLC link with an LLC2 link. (When local termination is the goal of using SLCS, SLCS may connect SDLC to SDLC or LLC2 to LLC2.)

- LLC2 links involve two addresses: one at each end of the link.
- SDLC links have only one address because the primary device does not need an address while the secondary devices must have unique addresses on one SDLC port to differentiate them from each other.

The basic rules for addressing different SLCS session sides are as follows (note that there are several actual examples involving hosts, FEPs, SLCS, and controllers later in this chapter.):

The LLC2 side:

- **Local MAC address:** this is the MAC address of the SLCS link station that is the local end of a given SLCS session. It is a virtual - or a *spoofed* address - in the sense that it is neither the MAC address of the router nor of any other particular device. However, it is the address that the LLC2 node on the far side of the LLC2 session associates with the device that resides on the far side of the SLCS session.
- **Remote MAC address:** this is the MAC address of the device on the far side of this LLC2 session (there could be LLC2 sessions on both Side A and Side B.). This is a real MAC address that designates an actual LLC2 device. When an LLC2 session involves two SLCSs, this address is also the virtual spoofed address of the far-end device.
- **XID values:** these need to match the values in the hosts' VTAM GEN file when the host is attached to a Token Ring LAN through a 3745 FEP. (In this case, the LLC2 session is between the SLCS router and the 3745 FEP.)

The SDLC side:

- **SDLC address when the router is set as a primary SDLC device:** the SDLC address is a remote address that designates the downstream, secondary SDLC device. This address must match the SDLC address set in the downstream device.
- **SDLC address when the router is set as a secondary SDLC device:** the SDLC address designates the router to the primary SDLC device.

➤ **Note:** When Group Polling is used, the SLCS Group Poll address must be the same for each SLCS SDLC secondary session participating in group polling scheme. This address must also match the corresponding value configured in the primary SDLC station (i.e., FEP).

WAN Interface Settings

Whichever side of the SLCS session is configured as the SDLC side is directly connected to the SDLC physical interface (a WAN port). The port number for this WAN interface is set in the SLCS Session Definition window. There are several WAN interface parameters that must be set correctly, including:

- *Physical Interface*, *Clock Source*, and *Speed* should be set according to the interface type you are using, whether the router generates or recovers the clock source, and the line speed.
- *Operational mode* must be set to SLCS PassThrough.
- *MaxTime in router* should be set to disabled.
- *T1/E1 Parameters* are not relevant to SLCS.
- *Point-to-point compression*, *Data inversion*, *Backup mode* and *DCE resynchronization* should all be disabled.
- Select either *Full duplex* or *Half duplex*.
- Set *Idling mode* to *Flags*, *Mark*, or *Mixed*. (For AS/400 configurations, use *Mixed* on the side adjacent to AS/400.)

Still referring to Figure 83, the non-SDLC side of the A-side/B-side session is automatically connected to the *Virtual SLCS Interface*, which is connected to the Virtual Port.

Host, FEP, SLCS and Controller Parameters for PU 2.0 Devices

Three typical configurations are shown in this section. Using SLCS in these configurations can involve setting parameters in the SNA host (VTAM), the FEP's NCP, SLCS, and the cluster controller configuration windows.

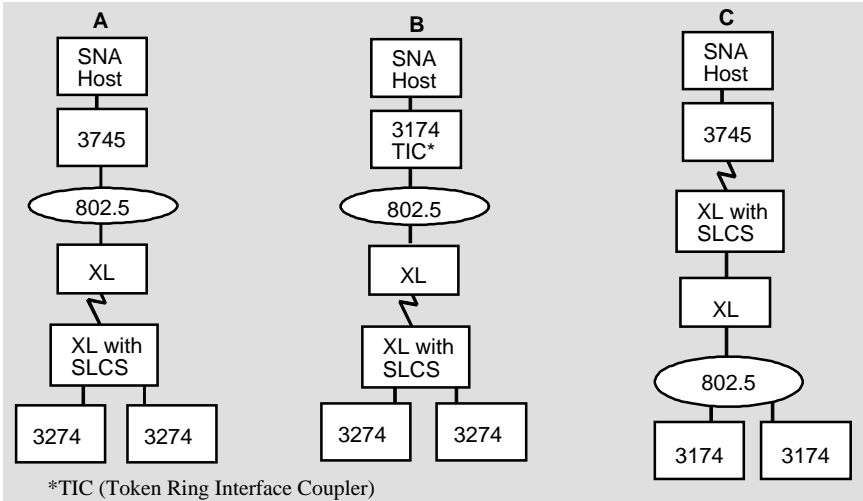


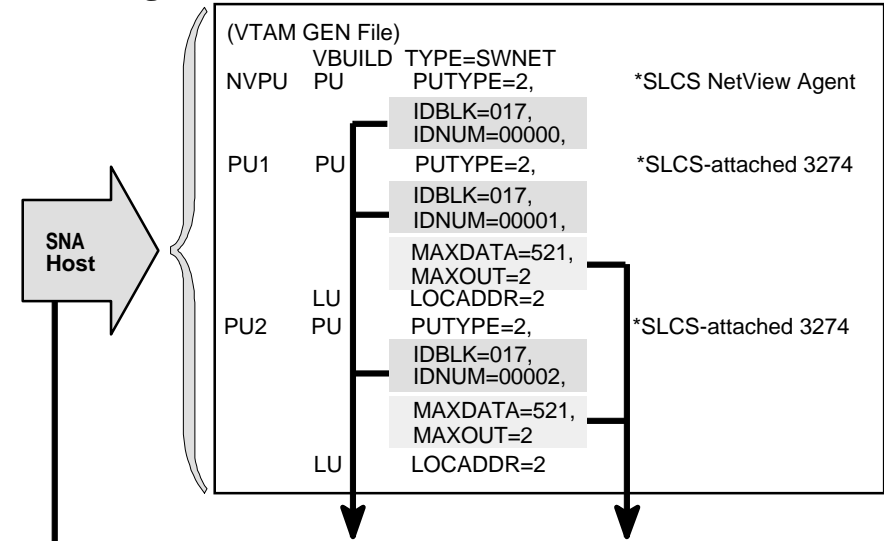
Figure 84. Three Typical SLCS configurations

Figure 84 shows the three typical configurations that are detailed in the next few pages.

- SNA configuration parameters are set in the host's VTAM GEN file
- NCP parameters are set in the FEP's NCP GEN File
- SLCS parameters are set in various SLCS windows (See *ClearSight Guide to Operations* for further details about the locations of these SLCS windows.)
- Controller parameters are set in the controllers' configuration screens. These screens are divided into "questions," which are expressed in the following form: Q900, for question 900; and Q380 for question 380.

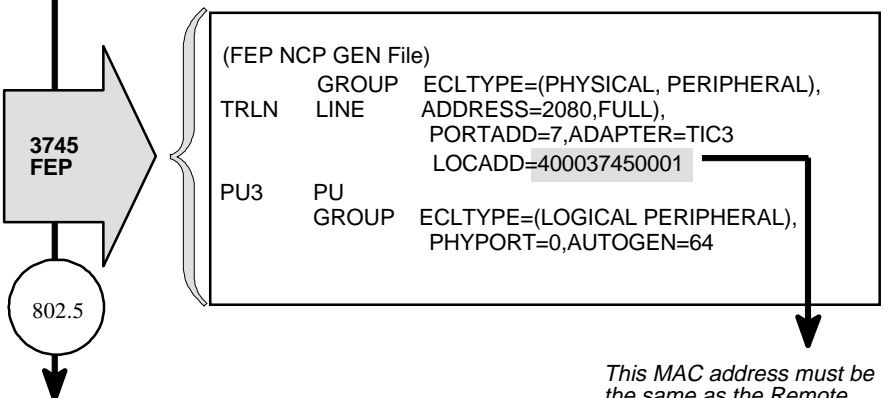
Not every parameter or variable in these files, screens and windows are discussed. Only the relevant variables and parameters are included here. Also note that the actual values used here (for instance addresses) may be arbitrary. It is simply noted here where values must match.

Configuration A



The IDBLK and the IDNUM values must be the same as the corresponding XID values in the NetView Agent and in the SLCS Session Definition windows.

MAXOUT must match the Window size and MAXDATA must match the Buffer size; both are set in the SLCS Tuning windows accessed from the SLCS Session Definition window.



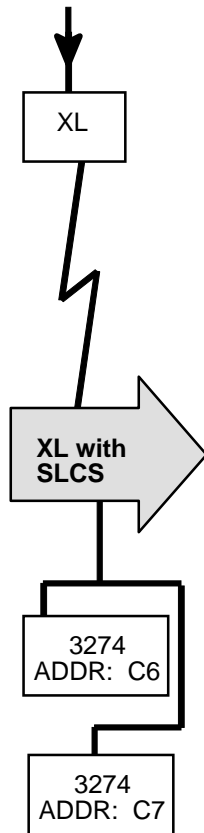
This MAC address must be the same as the Remote MAC address on the Token Ring side of the SLCS Session Definition windows.

(Network continued on next page)

(continued on next page)

Configuration A (continued from previous page)

(Network continued from preceding page)



SLCS Session Definition for <device> {session x}

Side A	Side B (Downstream)
<input type="radio"/> Start first PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input checked="" type="radio"/> Token Ring <input type="radio"/> SDLC	<input checked="" type="radio"/> Start first PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input type="radio"/> Token Ring <input checked="" type="radio"/> SDLC
XID - BLOCKID : 017	XID - BLOCKID :
XID - BLOCKNUM : 00001	XID - BLOCKNUM :
Local MAC : 40-00-20-00-00-05	<input checked="" type="checkbox"/> router on side B is SDLC primary
Local SAP : 04	Remote SDLC Addr: C6
Remote MAC : 40-00-37-45-00-01	Group poll : 0A
Remote SAP : 04	SLCS WAN port number : 3
<input type="button" value="Change"/> <input type="button" value="Refresh"/> <input type="button" value="Close"/> <input type="button" value="Tune..."/> <input type="button" value="Help"/>	

SLCS Session Definition for <device> {session x}

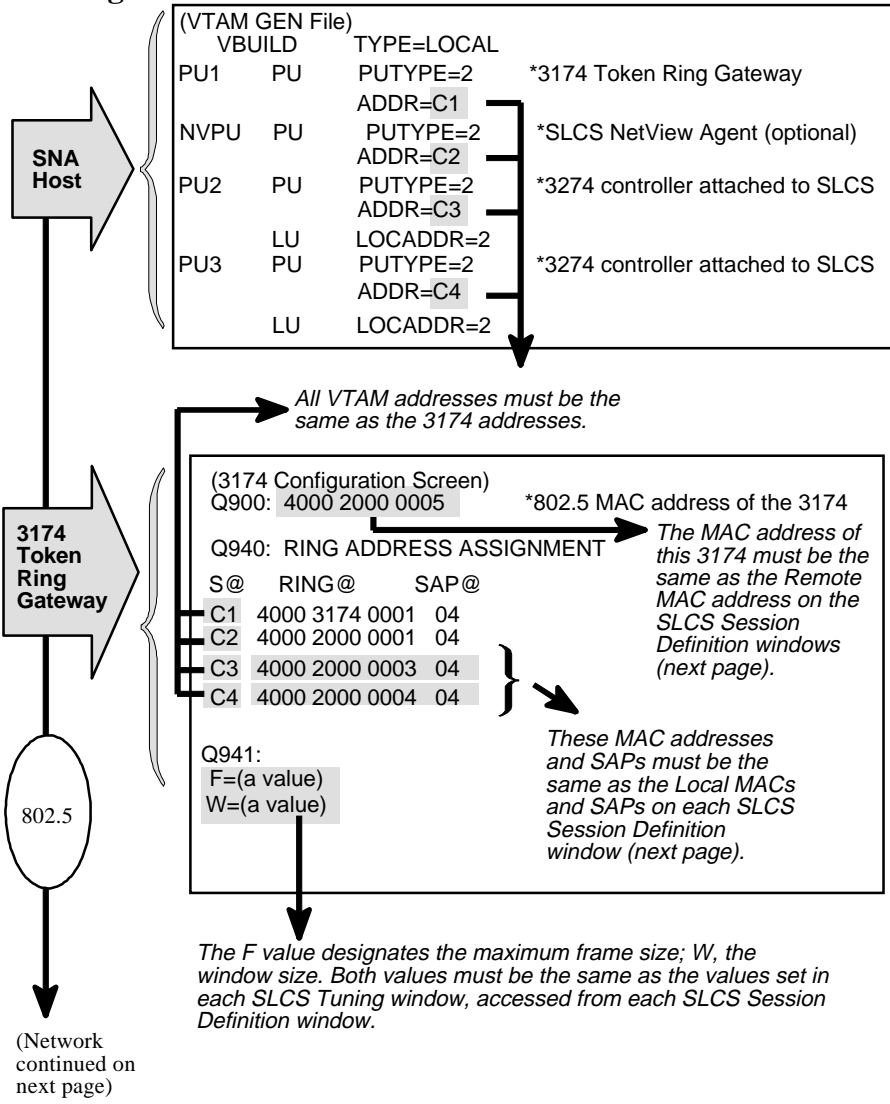
Side A	Side B (Downstream)
<input type="radio"/> Start first PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input checked="" type="radio"/> Token Ring <input type="radio"/> SDLC	<input checked="" type="radio"/> Start first PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input type="radio"/> Token Ring <input checked="" type="radio"/> SDLC
XID - BLOCKID : 017	XID - BLOCKID :
XID - BLOCKNUM : 00002	XID - BLOCKNUM :
Local MAC : 40-00-20-00-00-05	<input checked="" type="checkbox"/> router on side B is SDLC primary
Local SAP : 01	Remote SDLC Addr: C7
Remote MAC : 40-00-37-45-00-01	Group poll : 0A
Remote SAP : 04	SLCS WAN port number : 3
<input type="button" value="Change"/> <input type="button" value="Refresh"/> <input type="button" value="Close"/> <input type="button" value="Tune..."/> <input type="button" value="Help"/>	

① Each host-to-controller session defined in the HOST/VTAM and the FEP/NCP above must have its own SLCS session defined. The ID values shown above must map to the XID values in each SLCS session.

② These remote MAC addresses must map to the FEP's LINE LOCADD variable on the preceding page.

③ The Remote SDLC Address must be the same as the 3274's address. Because these sessions both use the same WAN port (that is, they are multidropped), the Remote SDLC addresses must be different.

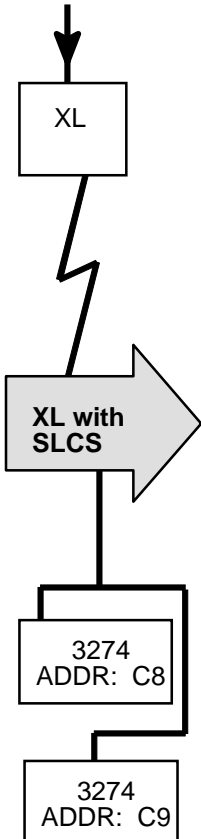
Configuration B



(continued on next page)

Configuration B (continued from previous page)

(Network continued from preceding page)



SLCS Session Definition for <device> {session x}	
Side A <input type="radio"/> Start first	Side B <input checked="" type="radio"/> Start first (Downstream)
PU type <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1	Access type <input type="radio"/> Ethernet <input checked="" type="radio"/> Token Ring <input type="radio"/> SDLC
XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/>	XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/>
Local MAC : <input type="text" value="40-00-20-00-00-03"/> ① Local SAP : <input type="text" value="04"/>	<input checked="" type="checkbox"/> router on side B is SDLC primary Remote SDLC Addr: <input type="text" value="C8"/>
Remote MAC : <input type="text" value="40-00-20-00-00-05"/> ② Remote SAP : <input type="text" value="04"/>	Group poll : <input type="text" value="0A"/> ③ SLCS WAN port number : <input type="text" value="3"/>
<input type="button" value="Change"/> <input type="button" value="Refresh"/> <input type="button" value="Close"/> <input type="button" value="Tune..."/> <input type="button" value="Help"/>	

SLCS Session Definition for <device> {session x}	
Side A <input type="radio"/> Start first	Side B <input checked="" type="radio"/> Start first (Downstream)
PU type <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1	Access type <input type="radio"/> Ethernet <input checked="" type="radio"/> Token Ring <input type="radio"/> SDLC
XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/>	XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/>
Local MAC : <input type="text" value="40-00-20-00-00-04"/> ① Local SAP : <input type="text" value="04"/>	<input checked="" type="checkbox"/> router on side B is SDLC primary Remote SDLC Addr: <input type="text" value="C9"/>
Remote MAC : <input type="text" value="40-00-20-00-00-05"/> ② Remote SAP : <input type="text" value="04"/>	Group poll : <input type="text" value="0A"/> ③ SLCS WAN port number : <input type="text" value="3"/>
<input type="button" value="Change"/> <input type="button" value="Refresh"/> <input type="button" value="Close"/> <input type="button" value="Tune..."/> <input type="button" value="Help"/>	

① Each host-to-controller session defined in VTAM and in the 3174 above must have its own SLCS session defined. The Local MAC and SAP values shown in the 3174's Q940 above must map to the Local MAC and SAP values in the SLCS sessions, shown here.

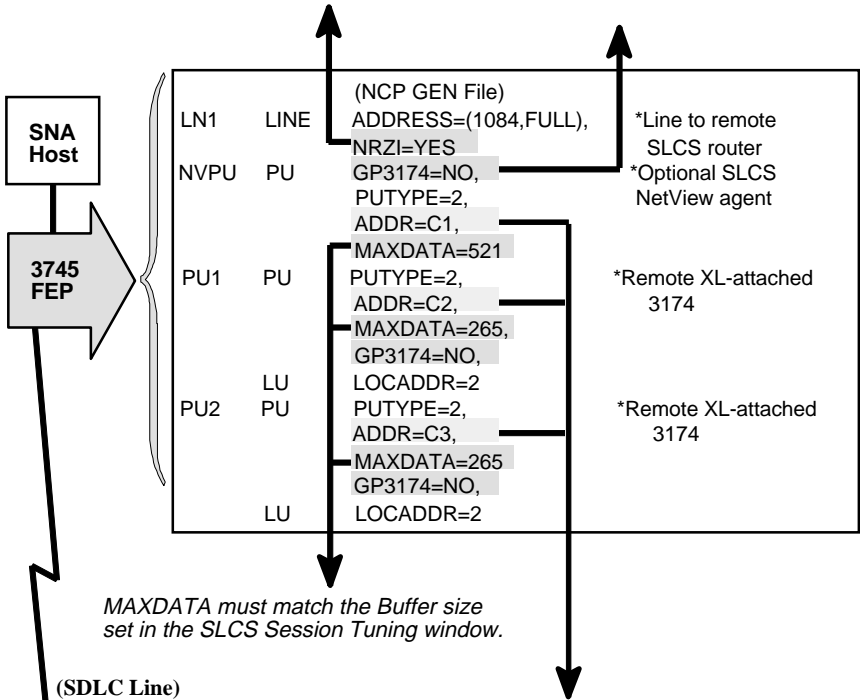
② These Remote MAC addresses must map to the 3174's Q900 address on the preceding page.

③ Note that the Remote SDLC Address is the same as the 3274 addressed. Also, because these sessions both use the same WAN port (that is, they are multidropped), the Remote SDLC addresses must be different.

Configuration C (first of three pages)

You may set NRZI to equal YES or No. The settings must match those in the ClearSight Interface Parameters window for the WAN port on the SLCS router. If NRZI=YES here, then the ClearSight field Encoding must be set to NRZI. If NRZI=NO here, then the ClearSight field Encoding must be set to NRZ.

This value sets the Group Polling parameters used by the NetView agent to poll the 3174s. When GP3174=NO, you do not need to match a parameter in SLCS. When GP3174 is set to a hex value, you must set the Group poll field on the SDLC side of each SLCS Session (you want to be group polled) to the same value. (See the SLCS windows on the next page.) All instances of the GP3174 variable must be set to the same value.

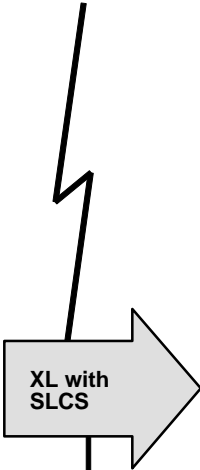


(Network continued on next two pages)

(continued on next page)

Configuration C (second of three pages)

(Network continued from preceding page)



(Network continued on next page)

SLCS Session Definition for <device> (session x)

Side A <input type="radio"/> Start first PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input type="radio"/> Token Ring <input checked="" type="radio"/> SDLC XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/> <input type="checkbox"/> router on side A is SDLC primary Remote SDLC Addr: C3 ① Group poll : <input type="text"/> ② SLCS WAN port number : 3	Side B <input checked="" type="radio"/> Start first (Downstream) PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input type="radio"/> Token Ring <input type="radio"/> SDLC XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/> Local MAC : 40-00-00-20-00-10 Local SAP : 04 ③ Remote MAC : 40-00-00-20-00-12 Remote SAP : 04
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Change Refresh Close Tune... Help

SLCS Session Definition for <device> (session x)

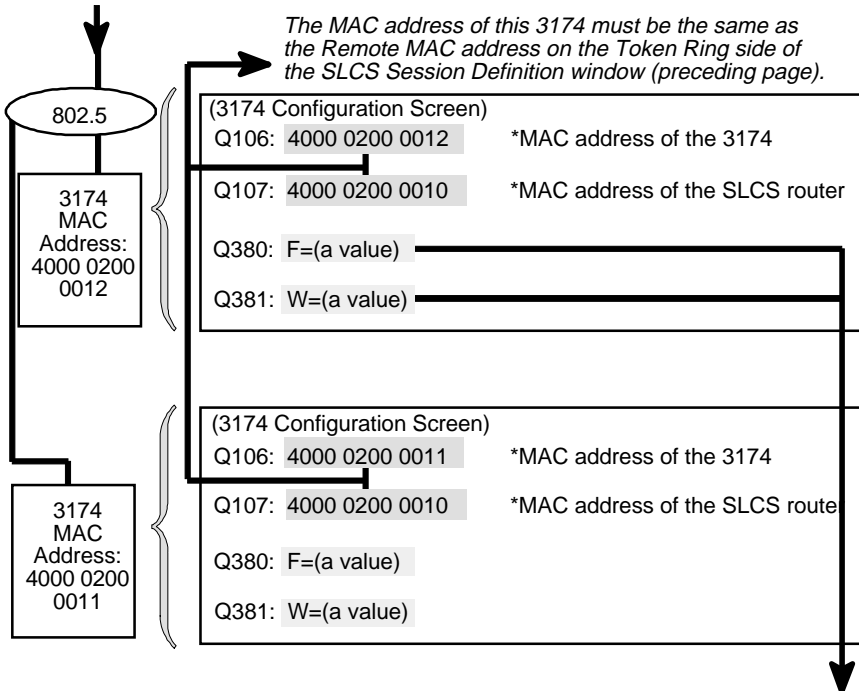
Side A <input type="radio"/> Start first PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input type="radio"/> Token Ring <input checked="" type="radio"/> SDLC XID - BLOCKID : 017 XID - BLOCKNUM : 0111 <input type="checkbox"/> router on side A is SDLC primary Remote SDLC Addr: C2 ① Group poll : <input type="text"/> ② SLCS WAN port number : 3	Side B <input checked="" type="radio"/> Start first (Downstream) PU type: <input type="radio"/> PU 1.0 <input checked="" type="radio"/> PU 2.0 <input type="radio"/> PU 2.1 Access type: <input type="radio"/> Ethernet <input checked="" type="radio"/> Token Ring <input type="radio"/> SDLC XID - BLOCKID : <input type="text"/> XID - BLOCKNUM : <input type="text"/> Local MAC : 40-00-00-20-00-10 Local SAP : 04 ③ Remote MAC : 40-00-00-20-00-11 Remote SAP : 04
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Change Refresh Close Tune... Help

- ① These Remote SDLC Addresses must be the same as the ADDR addresses in the NCP GEN File shown on the preceding page.
- ② These Group Poll values must equal the value set in the NCP GEN file for the SLCS NetView agent and for the PUs (shown on the preceding page). There are no values in the fields because the GP3174 variable in the NCP GEN file is set to NO.
- ③ The Remote MAC addresses must be the same as Q106 and the Local MAC addresses must be the same as Q107 on the 3174 configuration screens shown on the next page.

(continued on next page)

Configuration C (last of three pages)



The *F* value designates the maximum frame size; *W*, the window size. Both values must be the same as the values set in each SLCS Tuning window, child windows of each SLCS Session Definition window.

FID2 Segmentation, Multidrop and Group Polling

This section describes FID2 frame segmentation and Group Polling, two SLCS features that enhance SLCS functionality.

FID2 Frame Segmentation

SLCS supports the ability to segment large FID2 frames into smaller frames before passing them on. (FID2 designates the type of transmission header used on an SNA message.)

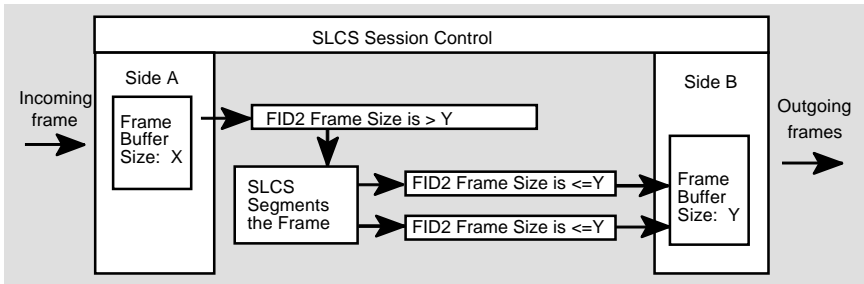


Figure 85. FID2 Frame Segmentation

Buffer sizes are set using ClearSight.

Multidrop and Group Polling

SLCS also supports multidrop configurations and group polling.

- *Multidrop* is the ability of the SLCS router to maintain sessions with multiple secondary SDLC devices over the same SDLC line.
- *Group Polling* is the ability of the FEP to query the SLCS router only once to obtain the information packets from downstream cluster controllers instead of querying each SDLC secondary device separately. This feature increases throughput over the SDLC WAN link.

Figure 86 shows the advantages of a multidrop configuration with Group Polling.

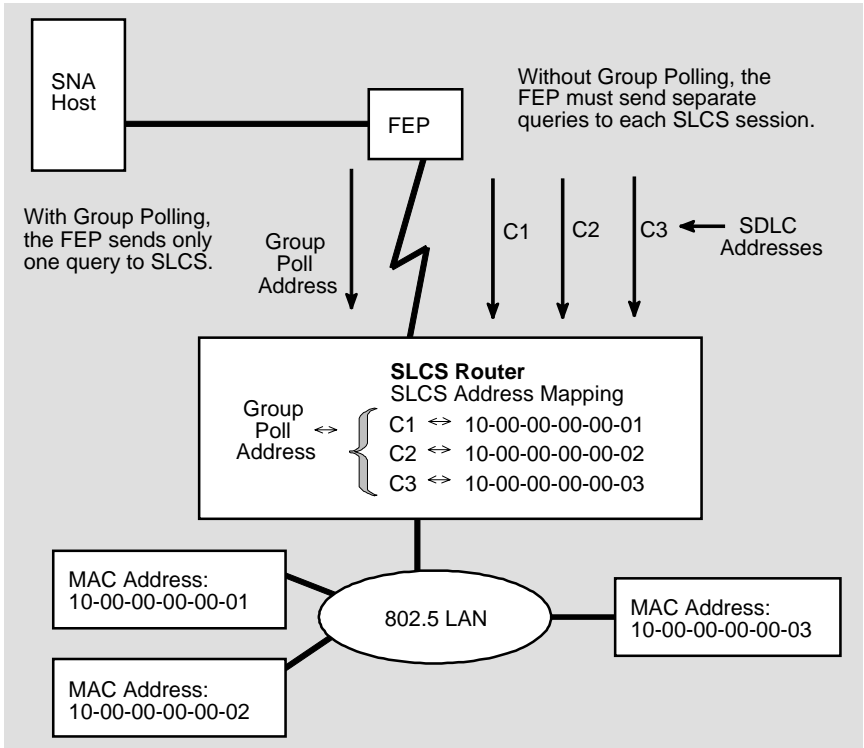


Figure 86. Multidrop and Group Polling

XID Identifiers During SLCS Session Establishment

One of the session establishment phases is the exchange of station identification information. Frames carrying such information are called XID frames. There are various formats of XIDs:

- XID 0 for PU 2.0 devices
- XID 1 for PU 1.0 devices
- XID 3 for PU 2.1 devices

All these XIDs contain the following fields.

- *BLOCKNUM* describes the device type (cluster controller, PC, etc.).
- *BLOCKID* identifies the specific device.

Some of the XIDs contain more information.

If the host gateway is a FEP, both *BLOCKNUM* and *BLOCKID* should be defined during SLCS setup.

NetView Support

Olicom implements a SLCS NetView agent for SLCS. The agent allows a remote console that is logged into the NetView application (residing on a host system) to manage individual SLCS sessions. The NetView agent also allows the router to alert the host of unusual conditions.

To participate in the NetView management system, the SLCS NetView agent must be defined both on the router (see ClearSight XL Configuration and Management) and in the VTAM/NCP GEN files. Regarding the VTAM/NCP GEN files, the agent must be defined as a dedicated downstream PU (DSPU) in the SNA host and in the host gateway or as a dedicated PU on the SDLC line. No LU definitions are required. Any definition that is similar to a 3174 Cluster Controller may be used.

To manage the SLCS sessions from a remote NetView console, you must use NetView's NCCF (Network Command and Control Facility) to issue the following command:

```
RUNCMD SP = (insert PU name), APPL=ILAN, (insert command syntax
from Table 16)
```

Thus, a legal command would be:

```
RUNCMD SP = PU1, APPL=ILAN, Display Session=(insert SLCS
session number)
```

For instance, entering this command from the NCCF facility at a NetView console instructs the SLCS NetView agent to display the configuration of the specified session.

The SLCS NetView agent responds to the commands listed in Table 15 by displaying one of the codes displayed in Table 16 on the remote terminal. Table 17 provides explanations of the types of NetView alerts.

Command Syntax	Description
DISPLAY SESSION=N (where N is the SLCS session number)	Displays the configuration of the specified session.
STOP SESSION=N	Stops the specified session.
START SESSION=N	Starts the specified session.

Table 15. NetView Agent Commands to the Router

Code	Message	Description
CCC002E	COMMAND NOT EXECUTED	Command has not been executed.
CCC011E	SYNTAX ERROR IN COMMAND	Syntax error in user command.
CCC101I	COMMAND EXECUTED	Command has been executed.
CCC112E	SESSION DOES NOT EXIST	Session ID is out of range.
CCC113E	SESSION ALREADY STARTED	This session is already started.
CCC114E	SESSION ALREADY STOPPED	This session is already stopped.
CCC099I	DISPLAY END	End of Display Session replies.

Table 16. NetView Agent Messages

Type	Alert Description: Probable Cause
LINE	No communication with device: SDLC COMM/REM node
RING	Link error: connection not established
CSMA/CD	Link error: connection not established

Table 17. Types of NetView Alerts

Sample Complex Topology

This is a complex topology in which some routers use SLCS to terminate sessions and some do not. This allows mixing of various LAN types and uses WAN transport for multiple data streams.

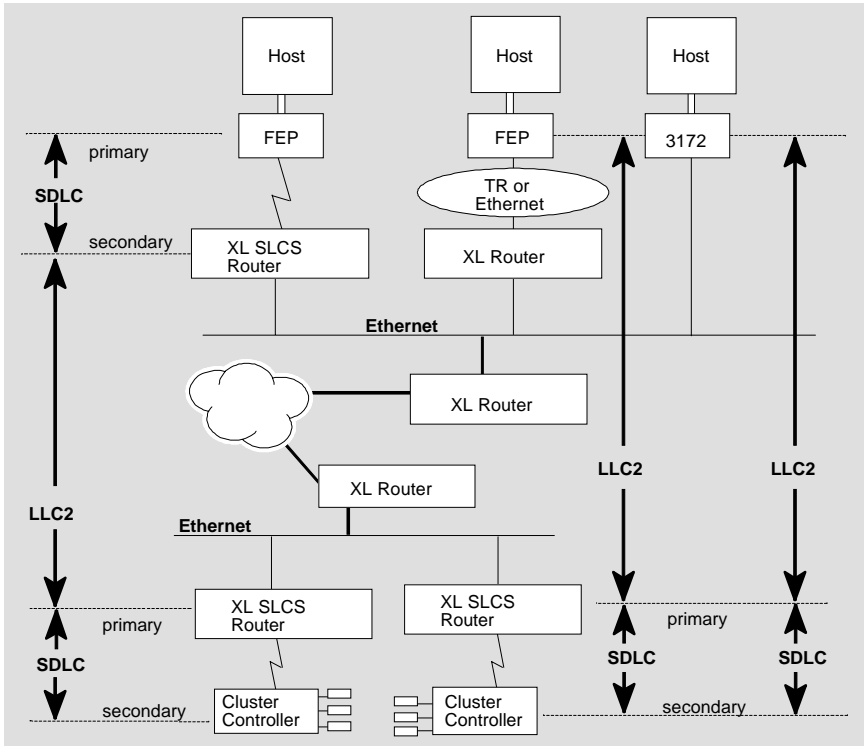


Figure 87. Complex SLCS Topology



14. DLSw

This chapter provides reference material that explains Olicom's DLSw (Data Link Switching) and provides key information useful during implementation and configuration.

Sections

- *Overview*
- *Typical DLSw Configuration*
- *Switch-to-Switch Protocol*
- *DLSw Partners Setup*
- *SNA Devices handling*
- *SDLC-attached SNA Devices handling*
- *NetBIOS Devices handling*
- *Benefits of Local Termination*
- *Flow Control*
- *DLSw Coexistence with Bridging and Routing*
- *Enabling/Disabling Traffic on given XL Ports*
- *Virtual Port Parameters*
- *DLSw Filters*
- *Priority Management*

Overview

Data link switching is a method for handling SNA and NetBIOS data traffic.

Olicom's implementation of DLSw is based on RFC 1795 and RFC 1434 standards.

DLSw makes two communicating end stations each appear adjacent to the other on a shared data link. The data link can be one of the following: LLC type-1 or LLC type-2 (on Token Ring and Ethernet), or SDLC. DLSw combines two data links by terminating each logically and relaying the data between them using TCP

as shown in Figure 88. A DLSw router appears to its local end stations as a collection of remote end station.

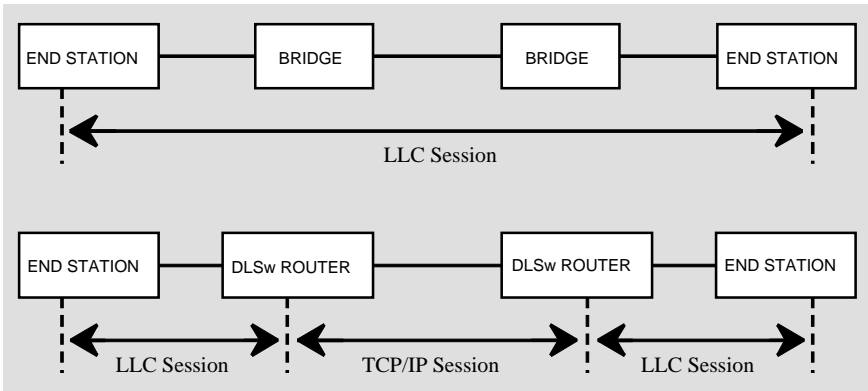


Figure 88. Normal bridging and DLSw

The following terms are used when describing DLSw:

- A *circuit* is an end station to end station association of two data links.

► **Note:** These two data links may run different data link layer protocols.

- A *transport connection* is a connection between two DLSw routers (partner routers). Transport connections are used for data transfer on active circuits and for carrying network control messages.

► **Note:** There are usually some intermediate routers on a physical path for data flowing on a transport connection, but these routers don't have to support DLSw. Data are simply forwarded by them.

DLSw is normally passive. When an end station (origin station) starts to send frames to another end station (target station), the following happens:

1. Searching for the destination station.

The DLSw router adjacent to the origin station starts searching for the destination station. It sends to its partners a message defined for locating remote resources. The partners search for the target on their local ports. If a partner finds the target it returns a special message.

During the searches routers retain information about locations of different resources. They use this database of locations in future searches when they need to find out which DLSw router is serving a given end station. DLSw makes use of caching to reduce the need for full broadcast searches as described in the following.

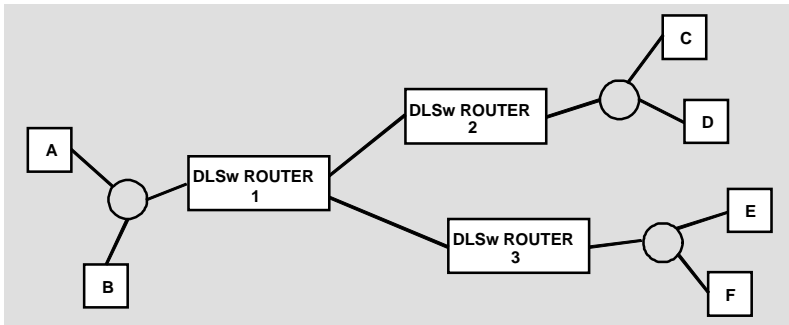


Figure 89. Example configuration

Suppose that end station A tries to communicate with end station E. DLSw router 1 sends a message to all its partners (2 and 3) to find out which of them can access station E. When router 1 determines that the target station is served by router 3 it caches this information. The next time any station adjacent to router 1 (A or B) tries to communicate with station E, router 1 will send a message only to router 3 rather than to all partner routers to see if the target is reachable.

2. Establishing a circuit.

If the origin station identifies the DLSw router adjacent to the target station it sends frames establishing a circuit between the stations via partner routers. Both partner DLSw routers create an internal database entry representing the circuit.

A circuit is identified by:

- local MAC address
- local SAP
- remote MAC address
- remote SAP

The most important circuit states are:

- *Disconnected* - when there is no circuit between a pair of end stations.
- *Circuit Established* - when two end stations exchange only datagram traffic.
- *Connected* - when the data links of the end station are set in connection oriented mode.
- *Disconnect Pending* - when a partner is waiting for acknowledgment to disconnect from its partner.

Typical DLsw configuration

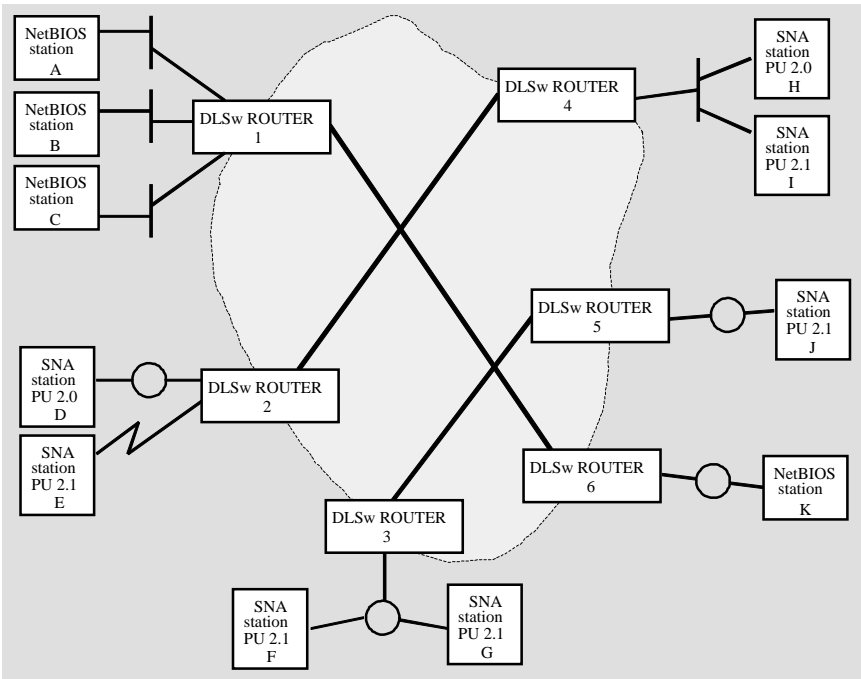


Figure 90. Example DLsw configuration

DLsw searches and normal data traffic can flow over the defined DLsw transport connections only. As shown in Figure 90, the topology of DLsw partners may not be fully meshed. A single network may support disjoint sets of partnerships.

Depending on the topology some end stations are able to communicate with others and some aren't. For example, end stations D and E are able to find and communicate with stations H and I via routers 2 and 4, but cannot reach station J. Stations from the same LAN also communicate via adjacent DLsw routers. For example, end station A reaches end stations B or C via router 1.

Data flowing on established circuits (for example, between stations D and H) is multiplexed with data for other established circuits (for example, between stations E and I) and also with search and control messages.

Switch-to-Switch Protocol

DLSw uses switch-to-switch protocol (SSP) to transfer data between partner nodes. The frame type sent by the end stations can be either an LLC (Ethernet or Token Ring) or SDLC frame. When such a frame from an origin station arrives at a DLSw router it is converted into the SSP frame format. The target DLSw makes the opposite transformation and sends it to the target end station as an LLC or SDLC frame (depending on the target station).

There are two SSP header formats:

- A longer header containing end station addresses used for searching and controlling circuits and for datagrams.
- A shorter header containing circuit identifiers used in messages carrying data.

SSP messages are delivered by TCP. They can be split across multiple TCP segments or combined into a single segment (as TCP is stream-oriented). The receiving DLSw reads bytes from TCP until the entire SSP message has been received and then processes the message. The length of the message is known from the length field in the SSP message header.

► **Note:** For more information on TCP please refer to chapter 6, *TCP* in this volume.

DLSw partner setup

It is up to the user which of the routers in a network can establish a transport connection with whom. For every DLSw node, the user defines a set of partners identified by their IP addresses.

As shown on Figure 90, DLSw routers are on the edge of an IP cloud interfacing to end stations on one side and to an IP network via SSP (switch-to-switch protocol) on the other. Transport connection topology determines which end stations are able to communicate. This means that not every pair of DLSw-served end stations is able to communicate. However, every router can transfer normal routed traffic to any other.

If a DLSw router doesn't manage to connect to any of its partners or if it loses the connection, it tries periodically to reach that partner (a connection may go down for example when one of the partners has been switched off).

After establishing a TCP connection between two routers, each of them sends some information about its identity and capabilities to the other. This message is called a *cap_exchange*. It contains the following information:

- Vendor ID, indicating whose software is running
- Version number of the DLSw standard supported
- An initialization value for the flow control algorithm

- A list of the LLC SAPs supported by the sender

The partner routers may also send the following information:

- A free-format text string to identify the version of the sending software
- The desired number of TCP connections
- A list of MAC addresses for SNA end stations local to the router
- A list of NetBIOS names local to the router
- Any vendor-defined capabilities

The receiver of a *cap_exchange* has to acknowledge this message, whether accepting or rejecting it. It uses the same type of messages, a *cap_exchange*. If the acknowledgment is negative the transport connection is taken down. If there is no reply for the *cap_exchange* the sender assumes that the partner is RFC 1424 - compliant.

After exchanging the routers' capabilities, the transport connection is ready to carry messages to search, control circuits and transfer data.

- **Note:** If for any reason a partner's capabilities change after the initial exchange, the partners can send *cap_exchange* messages again.

SNA Device Handling

There are four phases to every SNA connection via DLSw:

1. Search

After receiving a TEST frame from the origin station the DLSw router sends a *canureach_ex* message to all its partners or a specific one if it is already known as the router serving the target station. This message contains the addresses of the origin and target end stations. Partner routers that can reach the target station through one of their ports send back a message called *icanreach_ex*. The router which responds the first is regarded as the best route to reach the target station.

2. Establishment

When a DLSw router receives the first XID (exchange identifier) frame from the origin station it starts establishing a circuit.

- a. It sends a *canureach_cs* to the router that responded during the first phase.
- b. If this router is still able to reach the target it returns *icanreach_cs*.

- c. Then the origin router acknowledges it by sending a *reach_ack*.

These messages contain the circuit identifier. The DLSw routers then use a special *xidframe* message to transport SNA XID frames between routers. Finally, SNA protocol commands that are used to set mode are transported between partners using the SSP messages *contact* and *contacted*. At this point the circuit is fully connected.

3. Connected

During this phase, DLSw partners use *infoframe* messages to transport user data between end stations.

4. Disconnect

The SNA frames used to terminate a connection are passed between DLSw partners using *halt_dl* (halt data link) and *dl_halted* (data link halted) messages. A circuit also can be disconnected if an intermediate router fails.

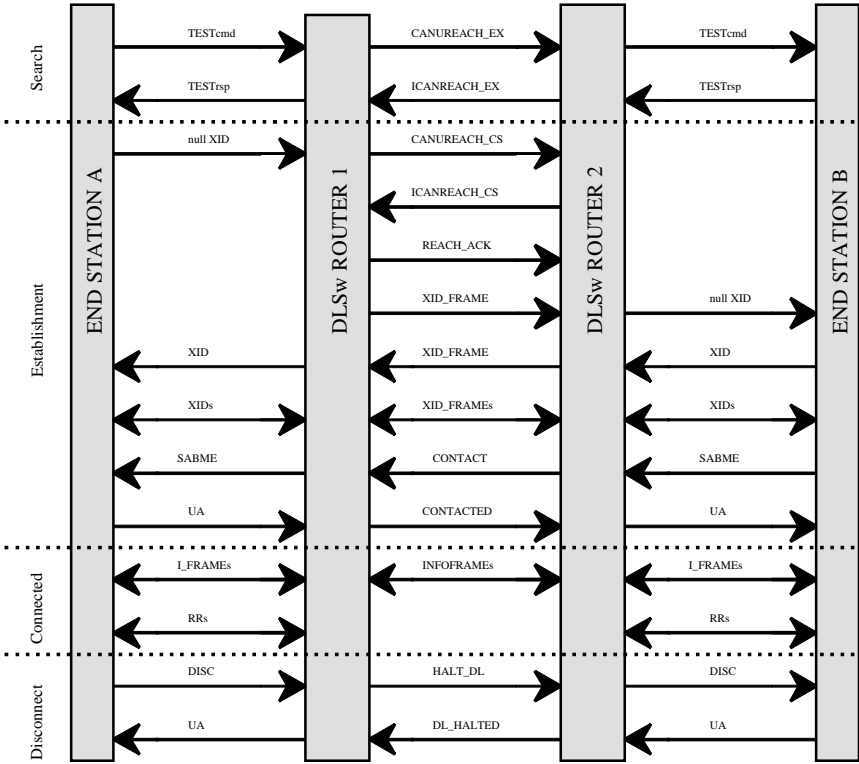


Figure 91. Connecting SNA devices

SDLC-Attached SNA Device Handling

SDLC-attached SNA devices are handled in a DLSw node using an intermediate XL. SDLC-to-LLC converter services process (SLCS) session. The SLCS session logically terminates the SDLC session on one side and pretends to be a normal LLC device on the other side. To do so, each SDLC device has to be specially configured within SLCS. It must contain at least the following data:

- device type: PU2.0 or PU2.1
- on the SDLC side: SDLC address; WAN port number; and SDLC role: Primary, Secondary or Negotiable
- on the LLC side: Access Type (Ethernet or Token Ring), device's MAC and SAP, and partner's MAC and SAP.

Using this arrangement, the SDLC device will behave like the one talking to a locally-connected SDLC partner device. The DLSw node, in turn, will act as if it were servicing a normal LLC-attached SNA device.

➤ **Note:** For more details, see chapter 13, *SLCS* in this volume.

NetBIOS Devices handling

There are five phases to every NetBIOS connection via DLSw:

1. Name Registration

When an application becomes active it is necessary to ensure that no other application is using the same name. When a DLSw router receives from an origin station a special NetBIOS frame (Add Name Query) used to check the name it forwards this frame to all its partners using a message called *netbios_anq*. The destination DLSw sends this to all its end stations. No answer means no name collision.

2. Name Search

When an application calls another application it sends a NetBIOS Name Query frame to find the end station with that application. DLSw forwards this frame, using a *netbios_nq_ex* message, to all its partners or a specific one if it is already known. The partners broadcast the frame on their LANs. If any end station responds it sends back a *netbios_nr_ex* message which is then forwarded by the origin DLSw to its LAN using a NetBIOS Name Recognized frame.

3. Establishment

To establish a connection the origin station sends a frame used to set mode. As in the SNA Establishment phase described previously, the DLSw routers send the following messages.

- a. The origin router sends a *canureach_cs* to the router that responded during the first phase.
- b. If this router is still able to reach the target it returns *icanreach_cs*.
- c. Then the origin router acknowledges it by sending a *reach_ack*.

To move a circuit into connected state the two routers exchange *contact* and *contacted* messages.

4. Connected

During this phase end stations exchange frames with user data. These frames are carried over the circuit by DLSw partners using *infoframes*.

5. Disconnect

The frames sent by end stations in order to end the session are forwarded by DLSw routers using *halt_dl* and *dl_halted* messages.

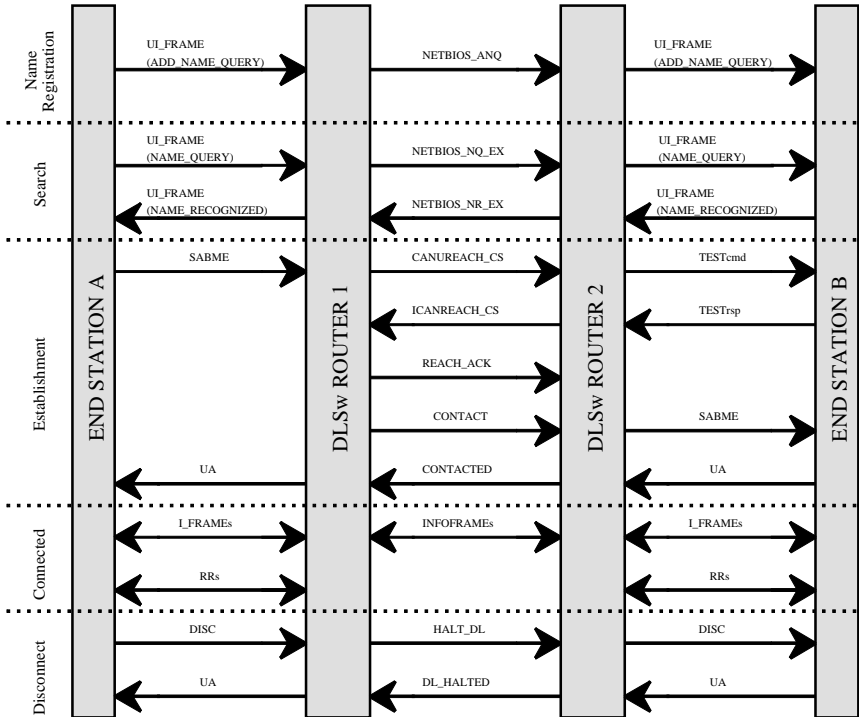


Figure 92. Connecting NetBIOS devices

Benefits of Local Termination

End stations communicate through DLSw routers adjacent to these stations. The routers pretend to their local stations to be the remote end stations. Connection oriented traffic is performed and acknowledged locally. To transport information traffic across the WAN DLSw routers use delay-insensitive switch-to-switch protocol (SSP).

Time-out Avoidance

In a normally bridged or routed network the time a frame needs to be passed between two stations may become very long. The more nodes there are on the path, the more time it takes to move along this path. Timeouts may occur when it takes longer than the time allowed. The situation is quite different when you use DLSw routers. Due to local termination and limiting the service traffic to the local network the delays are much smaller than in the case of normal bridging. In this way the problem of link time-outs is avoided.

Broadcast Avoidance

During a NetBIOS session an application may send some data to another application with a specific name or to a group of applications. Applications using group names send their data broadcasting frames to the target group address. These broadcasts can be easily limited when you use DLSw routers which do not forward frames to all partners. There are two sources of information taken into account when choosing the partner routers as the targets:

- NetBIOS name lists received during capabilities exchange.
- Information about associations between DLSw routers and NetBIOS names, cached during previous name searches and name registrations.

Flow Control

Congestion may occur when the throughputs of devices and data links along a circuit are different. A congested circuit may affect other circuits on the same transport connection. DLSw contains a windowing mechanism to solve this problem.

Messages are transported in *windows*. The number of messages a window can contain is determined by its size, which is set during the initial capability exchange. All circuits on a given transport connection use the same initial window size.

A router sends messages only if the receiver grants it special permission. If the receiver becomes congested it chooses one of the following solutions:

- withholding permission

- reducing the size of the window
- signaling its partner to stop sending immediately

If the congestion conditions stop, the receiver returns permission and increases the window to let the sender transmit more messages.

DLSw Coexistence with Bridging and Routing

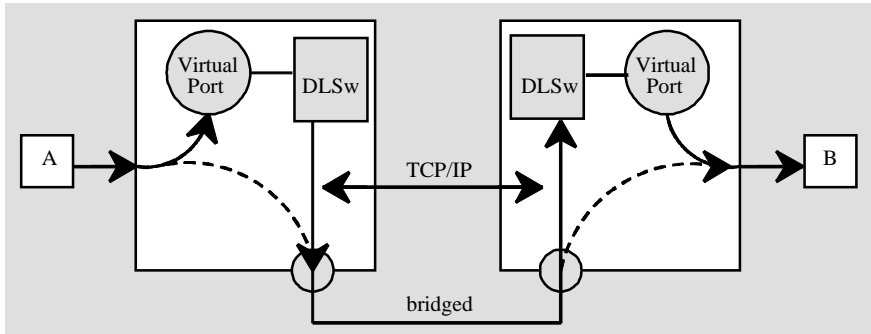


Figure 93. Bridging and Routing

When a frame reaches the router port its type is checked. If it is a NetBIOS or SNA frame it goes to DLSw through a virtual port and then a TCP connection is used to forward the frame. If it is neither NetBIOS nor SNA frame a bridge or route path is selected. The frame's type is recognized according to its SSAP field. If it is an SNA frame the SSAP field contains 04, 08, 0C... or A4 byte; if it's a NetBIOS frame it contains F0. In this way SNA and NetBIOS frames are prevented from being simultaneously bridged and passed via DLSw.

When a link between two DLSw nodes is a Dial on Demand link, unnecessary traffic on that link should be avoided. For routed traffic there are various methods of doing so, such as spoofing or reduced advertising. Bridged traffic can be either enabled, partially filtered, or completely disabled over that link.

Enabling/Disabling Traffic on given XL ports

SNA/NetBIOS Traffic

An XL router is in fact a b-router. That is, it routes routed traffic (based on the layer 3 protocol addresses) and bridges any other traffic (based on the layer 2 MAC addresses). When an XL router is acting as a DLSw node, you have to ensure that SNA or NetBIOS traffic will not be forwarded additionally via the bridging mechanism. To prevent such unnecessary forwarding you can specify which XL ports are designated to forward SNA or NetBIOS traffic exclusively via DLSw. When selected, the ports will not participate in bridging SNA or NetBIOS frames, but will cooperate with DLSw handling routines only. On a given port you can specify SNA only, NetBIOS only, or both types of traffic as “DLSw -handled”.

This setting can be done using the Port Parameters ClearSight screen, or with a SET PORT console command. By default, all ports have SNA & NetBIOS traffic handling disabled (meaning such traffic will be bridged, not handled by DLSw). This setting is recorded in nonvolatile memory and restored after device reboot.

Bridged Traffic

As mentioned previously, you may want to switch port-level bridging on or off to accommodate Dial on Demand. This can be done using the **Port Parameters** ClearSight screen or the SET PORT console command. The setting is recorded in a nonvolatile memory and updated after the device reboot. By default all ports have bridging enabled unless bridging is disabled globally. (Bridging can be disabled globally by selecting a Bridging/Routing mode of *none* in the **Bridging and PIR** ClearSight dialog, but be aware that this setting would effectively disable DLSw on that node.)

Virtual Port Parameters

The virtual port is an internal XL router arrangement designed to pass LLC frames between XL bridging routines and the DLSw. The DLSw uses that port to receive any LLC frames from real LAN ports and to transmit LLC frames, which are subsequently directed by XL bridging routines to real LAN ports. For Ethernet frames the virtual port is transparent; for Token Ring frames it looks like a Token Ring segment. Therefore, in order to pass TR frames correctly, it must be assigned a unique LAN segment number.

From the management point of view, the virtual port behaves exactly like a normal port with one exception: it is not mapped to any real XL interface. Instead, it is created dynamically when DLSw (or another application within an XL router using it) is started and removed if no longer needed.

DLSw Filters

To control the traffic on your DLSw routers, filters have been implemented. The filters affect both connection-oriented and connectionless traffic (UI frames only).

You can define three types of filters:

- **SAP** filters
- **MAC address** filters (for SNA traffic only; they have no impact on NetBIOS sessions)
- **NetBIOS name** filters

The last two may be in one of two classes:

- **Local** (from the side of the local station)
- **Remote** (from the side of the remote station)

Each combination of these filters can be:

- **Positive** - if frames matching the filter are excluded from passing through DLSw
- **Negative** - if frames matching the filter are the only frames passing through DLSw

You can specify whether a filter of a given type is positive or negative only while DLSw is down. Once you start it, all filters of that type that you define later will be either positive or negative depending on your earlier specification.

SAP Filters

SAP filters let you control which LLC frames will be transmitted via the DLSw router. Only the frames having SAPs included in SAP lists will be transmitted. Both DSAPs and SSAPs are checked.

There are two types of DSAP lists:

- **Global SAP list**, serving as a default enabled DSAP list for traffic directed to a DLSw partner.
- **Specific SAP list**, serving as an enabled DSAP list for traffic directed to the DLSw partner via a specific transport connection. Initially this list is preset to the values from the Global SAP list.

The DSAP list effective for the connection is sent to the partner DLSw node during the *cap_exchange* phase. The DSAP lists may have an associated priority value to be used for transmission to a partner DLSw node (see *Priority Management* on page 202).

The SSAP list is retrieved either from the SAP list received from a partner DLSw node during the *cap_exchange* phase, or is set equal to the local DSAP list effective for this connection (when the partner is a pre-RFC 1795 DLSw node).

MAC Address Filters

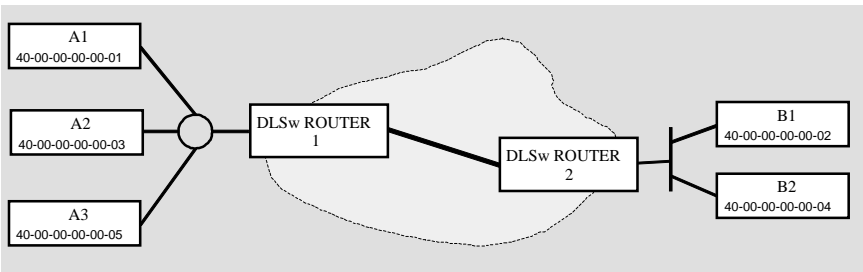
MAC address filters let you select a list of MAC source and destination addresses for which the UI LLC frames will or will not be passed or the LLC session will or will not be established through the DLSw router.

For a filter to take effect for a session, it must be defined before establishing that session.

To define a filter, enter the following information:

- Source MAC address and source mask
- Destination MAC address and destination mask
- Whether the filter is local or remote

The mask determines which bits of the given address are to be compared with the address in the frame. If you use FF-FF-FF-FF-FF-FF as the mask, the entire address will be compared.



Example:

If you want to filter the traffic from station A1 to station B1, use the following filter in Router 1:

Local MAC	Remote MAC	Type <input checked="" type="radio"/> Local <input type="radio"/> Remote
40-00-00-00-00-01	40-00-00-00-00-02	
Local Mask	Remote Mask	
FF-FF-FF-FF-FF-FF	FF-FF-FF-FF-FF-FF	

After applying this filter station A1 will not be able to contact station B1, though station B1 will be able to initialize a connection with station A1.

You can disable any sessions between station A1 and station B1, regardless of where the session was initialized, in one of two ways:

- Define an additional local filter in Router 2

Local MAC 40-00-00-00-00-02	Remote MAC 40-00-00-00-00-01	Type <input checked="" type="radio"/> Local <input type="radio"/> Remote
Local Mask FF-FF-FF-FF-FF-FF	Remote Mask FF-FF-FF-FF-FF-FF	

or:

- Define a remote filter in Router 1:

Local MAC 40-00-00-00-00-01	Remote MAC 40-00-00-00-00-02	Type <input type="radio"/> Local <input checked="" type="radio"/> Remote
Local Mask FF-FF-FF-FF-FF-FF	Remote Mask FF-FF-FF-FF-FF-FF	

Example:

If you want to enable traffic only from station A2 to station B2, define the following negative filter in Router 1:

Local MAC 40-00-00-00-00-03	Remote MAC 40-00-00-00-00-04	Type <input checked="" type="radio"/> Local <input type="radio"/> Remote
Local Mask FF-FF-FF-FF-FF-FF	Remote Mask FF-FF-FF-FF-FF-FF	

After applying this filter stations A1 and A3 will not be able to contact any other, but station B1 will be able to establish a connection. There are two ways to disable this:

- Define an additional local filter in Router 2

Local MAC 40-00-00-00-00-04	Remote MAC 40-00-00-00-00-03	Type <input checked="" type="radio"/> Local <input type="radio"/> Remote
Local Mask FF-FF-FF-FF-FF-FF	Remote Mask FF-FF-FF-FF-FF-FF	

or:

- Define a remote filter in Router 1:

Local MAC 40-00-00-00-00-03	Remote MAC 40-00-00-00-00-04	Type <input type="radio"/> Local <input checked="" type="radio"/> Remote
Local Mask FF-FF-FF-FF-FF-FF	Remote Mask FF-FF-FF-FF-FF-FF	

Now only sessions between A2 and B2 can be established. If you change your mind and want sessions between A3 and B2 to be enabled as well, define more filters in the same way as described above.

Example:

If you want to filter frames coming from the address range: 40-00-00-00-00-00 to 40-00-00-00-00-0F, then instead of defining 16 filters you can define just one specifying the proper mask.

Local MAC 40-00-00-00-00-00	Remote MAC 40-00-00-00-00-01	Type <input checked="" type="radio"/> Local <input type="radio"/> Remote
Local Mask FF-FF-FF-FF-FF-F0	Remote Mask FF-FF-FF-FF-FF-FF	

Warning If you choose to use negative MAC address name filters but you don't define any MAC address filters, no SNA session will be established.

NetBIOS Name Filters

NetBIOS name filters let you select the NetBIOS names for which the NetBIOS datagrams will or will not be passed or the NetBIOS session will or will not be established through the DLSw router.

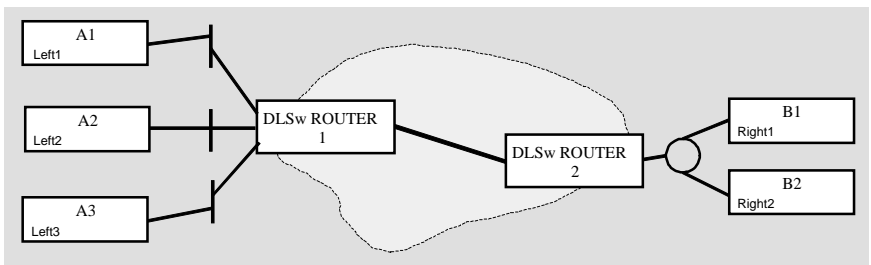
For a filter to take effect, it must be defined before establishing the session to which the filter is applied.

To define a filter, enter the following information:

- Source name
- Destination name
- Whether the filter is local or remote

NetBIOS names can have either 15 or 16 (default) characters. You choose the length of the names before starting DLSw. If the name is shorter than 15 (or 16) bytes, you have to fill the empty characters with spaces. You can use wildcards in the names: ? replaces any character at this position, * (which can be placed as the last character only) replaces any substring starting at this position. In this way you can specify a group of names using a single string.

In NetBIOS names all 16 (or 15 leading) bytes are significant. They may include non-printable characters, including NULL (0x00) characters. The ClearSight-entered NetBIOS names are case-sensitive (letters are not converted automatically to uppercase); the rest of the line is padded with spaces.



Example:

If you want to filter the traffic from station A1 to station B1, use the following filter in Router 1:

Local name	Remote name	Type
Left1	Right1	<input checked="" type="radio"/> Local <input type="radio"/> Remote

Now station A1 is not able to contact station B1, though station B1 can contact any station. To disable connections between these two stations in both directions, apply one of the following solutions:

- Define an additional filter in Router 2.

Local name	Remote name	Type
<input type="text" value="Right1"/>	<input type="text" value="Left1"/>	<input checked="" type="radio"/> Local <input type="radio"/> Remote

or:

- Define a remote filter in Router 1.

Local name	Remote name	Type
<input type="text" value="Left1"/>	<input type="text" value="Right1"/>	<input type="radio"/> Local <input checked="" type="radio"/> Remote

Example:

If you want to enable only sessions between stations A2 and B2, use the following negative filter in Router 1:

Local name	Remote name	Type
<input type="text" value="Left2"/>	<input type="text" value="Right2"/>	<input checked="" type="radio"/> Local <input type="radio"/> Remote

If you want these two stations to be the only ones that can contact one another you should add one of the filters shown below:

- A local filter in Router 2.

Local name	Remote name	Type
<input type="text" value="Right2"/>	<input type="text" value="Left2"/>	<input checked="" type="radio"/> Local <input type="radio"/> Remote

or:

- A remote filter in Router 1.

Local name <input type="text" value="Left2"/>	Remote name <input type="text" value="Right2"/>	Type <input type="radio"/> Local <input checked="" type="radio"/> Remote
---------------------------------------------------------	-----------------------------------------------------------	---------------------------------------------------------------------------------------

➤ **Warning** If you choose to use negative NetBIOS name filters but you don't define any NetBIOS name filters, no NetBIOS session will be established.

Priority Management

To optimize the traffic handling over DLSw connections, you can assign a particular priorities to specific transport connections, SAPs, or even selected frames. There are four priority levels:

1. Low
2. Medium
3. High
4. Highest

There are four ways to assign priorities:

1. Define a prioritizer.

A prioritizer is an address specifier that is defined like a MAC or NetBIOS filter, and that acts as a local filter. If a frame matches this address specification, its priority is set to the value you set for this prioritizer.

- If you are running a NetBIOS session and a NetBIOS prioritizer has not been defined, the MAC prioritizer is checked as well. If you are running an SNA session, only the MAC prioritizer is checked.
- If more than one prioritizer matches, the first one (alphabetically) is selected to enforce the priority.

In the example below when a frame from station 40-00-00-00-00-00 to station 40-00-00-00-00-01 passes the router, its priority is set to High.

Local MAC 40-00-00-00-00-00	Remote MAC 40-00-00-00-00-01	Type <input type="radio"/> Local <input type="radio"/> Remote <input checked="" type="radio"/> Prioritizer
Local Mask FF-FF-FF-FF-FF-FF	Remote Mask FF-FF-FF-FF-FF-FF	

2. Assign priorities to SAPs on a specific transport connection.

You can define a list of SAP numbers for a transport connection and assign those SAPs particular priority values. Figure 94 shows an example of how priority values might be assigned to specific SAP numbers on a transport connection.

SAP	Priority
00	Undefined
04	High
08	High
0C	Undefined
10	Low
14	Low
18	High
1C	Medium
20	Low

Figure 94. Priority values assignment to specific SAP numbers.

3. Assign a priority to a transport connection.

You can assign a transport connection a priority value which will be the priority for all circuits over this transport connection.

4. Set the default priority value.

The default priority value will be used when none of the above methods has been applied.

If more than one of the above definitions has been used, they are processed in the following order:

1. Default priority
2. TC priority
3. SAP priorities on TC
4. Prioritizer

The last processed item enforces the priority value used for transmission of a frame.

It is also possible to allocate for each priority value a fraction of the throughput available on a transport connection. In this way, you can guarantee sufficient relative throughput for a selected application based on the assigned priority value.



Index

A

Addressing

- classes 7
- IP 6
 - special filters 20
- IP conventions 19
- IP network numbers 6
- IP notations 19
- IP rules 19
- Secondary IP 8
- subnets 8
- variable length subnets 9

B

Best Route Determination

- IPX Routing 83

BGP

- advantages over EGP 55
- definition 55
- external peer 56
- features 55
- generating update messages 57
- internal peer 56
- IP routing table 56
- management 55
- message types 57
- overview 55
- specifying Export policy 57
- specifying Import policy 57
- terminology and concepts 60
- use of TCP 56

BOOTP

- virtual port and 126

Border Gateway Protocol

- See BGP

Broadcasting

- IP 12

D

Data Link Layer 113

Data link protocols

- LLC 160
- SLCS, data link protocols. 160

Data Link Switching

- DLSw 181

Discovery Shortest Path First

- See DSPF

DLSw

- circuit 182
 - establish 183
- configuration 184
- filters 195
 - MAC address 196
 - NetBIOS name 199
 - SAP 195
- flow control 192
- local termination 192
- NetBIOS devices 189
- overview 181
- partner 185
- priority 202
- SNA devices 186
- SSP 185
- TCP connection 193
- transport connection 182
 - establish 185
- virtual port 193, 194
- virtual port and 130
- windows 192

DSPF

- definition 4, 111

E

Export policy

- parameters 58

F

Filters

- IP 11

Frame format

- IPX packet 81, 89

G

global IP address

- replaced by virtual port 125

global IP mask

- replaced by virtual port 125

index-2

- I
- IGP
 - purpose 56
- Import policy
 - parameters 57
- Import/Export Policies
 - IP 15
- Interior Gateway Protocol
 - IGP 56
- Internet Protocol
 - See IP
- Internetwork Packet Exchange
 - See IPX Routing
- IP
 - addressing 6
 - classes 7
 - conventions 19
 - network numbers 6
 - notations 19
 - rules 19
 - subnets 8
 - variable length subnets 9
 - broadcasting 12
 - default gateway
 - replaced by virtual port 125
 - filters 11
 - network considerations 19
 - special filters examples 21
 - overview 5
 - RIP 25
 - Route Import/Export Policies 15
 - routing table
 - virtual port and 126
 - special filters 20
 - technical discussion 6
 - virtual port
 - BOOTP 126
 - IP host station 122
 - routing table 126
 - TCP/IP broadcast resolution 126
- IPX
 - Default Route
 - IPX Routing 97
 - Routing
 - IPX Default Route 97
- IPX Routing
 - best route determination 83
 - configuration example 105
 - configuration rules 109
 - features 77
 - IPX protocol 81
 - network considerations 105
 - Routing Information Protocol 86
 - Service Advertising Protocol 87
 - SPX Protocol 89
 - technical discussion 81
 - topology restrictions 109
- L
- LLC
 - LLC and SDLC compared 160
 - LLC1 and LLC2 compared 160
 - LLC2 and SDLC compared 160
- Loops
 - protocol independent routing 119
- M
- MAC sublayer 111
- N
- NetBIOS
 - interaction with IPX 81
- NetView agent
 - SLCS
 - SLCS, NetView agent. 178
- Network Layer 81, 111
- Novell IPX
 - protocol
 - IPX Routing 77
- O
- Open Shortest Path First
 - See OSPF
- OSPF 5, 35, 49, 113
 - AS 47
 - AS external link advertisement 45
 - authentication types 46
 - autonomous system 47
 - common header 38
 - configuring CFG 49
 - hierarchical architecture 47
 - ISO reference model 37
 - link state protocol 37
 - link-state advertisement header 40
 - link-state advertisement packets 40
 - network link advertisement 43
 - packet encoding 38
 - router link advertisement 41
 - routing areas 38
 - summary link advertisement 44
 - supported features 35
 - supported topologies 48
 - terminology 49

P

- PassThrough
 - configuration examples 148
 - features 145
 - general mode 146
 - general point to point 149
 - local multidrop 150
 - network considerations 148
 - SDLC point to point 149
 - SDLC-specific mode 146
 - technical discussion 146
 - topology restrictions 150
 - virtual multidrop 150
- PIR Cloud 111, 113
- Point to point
 - numbered 10
 - unnumbered 10
- Protocol Independent Routing (PIR) 111
 - DSPF Basics 113
 - DSPF/Source Routing Interactions 113
 - features 111
 - network considerations 117
 - PIR Areas 116
 - PIR cloud 111, 113
 - Selecting the DSPF Version 114
 - Source Routing and LAN Segment Addressing 115
 - technical discussion 113
 - topology restrictions 118
 - Version 1 configuration example 117
 - Version 2A configuration example 117
 - with source routing 115
- PU types
 - SLCS, PU types. 152

R

- RIP 5
- Routing Information Protocol
 - IPX Routing 86

S

- SDLC
 - SLCS, data link protocols. 161
- Secondary IP
 - addressing 8
- Service Advertising Protocol
 - IPX Routing 87
- SLCS
 - addresses 166
 - bridge modes 165
 - source routing 165
 - source routing transparent 165
 - transparent 165
 - configurations
 - local termination 155
 - SLCS and non SLCS 154
 - connecting to processes and ports 164, 167
 - data link protocols
 - LLC 160
 - LLC1 and LLC2 compared 160
 - SDLC and LLC compared 160
 - SDLC and LLC2 compared 161
 - feature summary 152
 - FID2 frame segmentation 176
 - frame conversion, SDLC to LLC 163
 - group polling 176
 - local termination 155
 - LLC2-LLC2 156
 - SDLC-LLC2-SDLC 155
 - MAC addresses
 - local 166
 - remote 166
 - multidrop 176
 - NetView agent 178
 - commands 179
 - messages 179
 - overview 152
 - parameters 166
 - PIR 165
 - port 164
 - port 18 164, 167
 - PU types
 - 1.0 152, 158, 159
 - 2.0 152, 153
 - 2.1 152, 158
 - overview 158
 - SDLC
 - primary/secondary nodes 161
 - SDLC addresses
 - primary router 166
 - secondary router 166
 - session sides 166
 - setting SLCS and SNA parameters 168
 - SNA
 - controller parameters 168
 - NCP parameters 168
 - SLCS configuration 168
 - VTAM parameters 168
 - time out avoidance
 - timeout avoidance 155

index-4

- topologies
 - PU 1.0 159
 - PU 2.0 153
 - LLC2 to SDLC 153
 - SDLC to LLC2 154
 - PU 2.1 158
- virtual port and 130
- WAN port 167
 - configuring 167
 - clock source 167
 - disabled parameters 167
 - maxtime in router 167
 - operational mode 167
 - physical interface 167
 - speed 167
 - T1/E1 167
 - XID parameters 166
- SNA Link Conversion Services
 - SLCS 151
- source routed bridging
 - virtual port and 128
- Source routing
 - IPX Routing 110
 - PIR 115
- Source Routing and LAN Segment Addressing
 - PIR 115
- SPX Protocol
 - IPX Routing 89
- SRT
 - IPX Routing 110
- SSP
 - Switch-to-Switch Protocol 185
- Switch-to-Switch Protocol 185

T

- TCP 63
 - as used by BGP 56
 - checksum 71
 - closing a connection 69
 - connections and ports 65
 - establishing a connection 69
 - finite state machine 72
 - glossary 74
 - header format 67
 - maximum segment size 70
 - overview 63
 - passive and active opens 69
 - segment format 67
 - sliding window 66
 - time-outs and retransmissions 67
- TCP/IP 5
- TCP/IP broadcast resolution
 - virtual port and 126

- Transmission Control Protocol
 - See TCP
- transparent bridging
 - virtual port and 128

V

- virtual port 121
 - BOOTP and 126
 - DLSw and 130
 - IP
 - simultaneous bridging and routing 122, 129
 - IP host station 122
 - IP routing table and 126
 - IP topology guidelines 127
 - overview 121
 - SLCS and 130
 - TCP/IP broadcast resolution and 126
 - technical discussion 122
 - topology guidelines 127
 - source routed bridging 128
 - transparent bridging 128



Olicom A/S

Nybrovej 114
2800 Lyngby
Denmark

Tel: (+45) 45 27 00 00
Fax: (+45) 45 27 01 01

Olicom Africa

Johannesburg Office:
6th Floor, Nedbank House
12 Fredman Drive
Sandown, Sandton

P.O. Box 785136
2146 Sandton

Tel: (+27) 11 784 8990
Fax: (+27) 11 784 9090

Cape Town Office:

4 Docav Road
Constantia
Cape Town

P.O. Box 6196
8012 Roggebaai

Tel: (+27) 21 758 525
Fax: (+27) 21 758 527

Olicom Australia

Level 6
73-79 Walker Street
North Sydney, NSW 2060

Tel: (+61) 2 9955 1755
Fax: (+61) 2 9955 8488

Olicom Austria

Mariahilfer Strasse 103/2/1/42A
1060 Vienna

Tel: (+43) 1 597 3131-0
Fax: (+43) 1 597 3131-31

Olicom Benelux

Bolduc Office Centre
Utopialaan 35-N
5232 CD 's-Hertogenbosch

Tel: (+31) 73 6 49 15 46
Fax: (+31) 73 6 49 15 45

Olicom, Inc.

1680 North Prospect Drive
Richardson, TX 75081

Tel: (+1) 972 907 4600
Fax: (+1) 972 671 7525

Olicom France

Immeuble Plein Ouest
177, Avenue G. Clemenceau
92024 Nanterre cedex

Tel: (+33) 01 41 91 17 17
Fax: (+33) 01 41 91 17 00

Olicom Germany

Frankfurt Office:

Hessenring 13a
64546 Mörfelden

Tel: (+49) 06 105 2892-0
Fax: (+49) 06 105 2892-10

Munich Office:

Stefan-George-Ring 29
81929 Munich

Tel: (+49) 89 993 936-0
Fax: (+49) 89 993 936-27

Olicom Ibérica

C/Basauri, 17 - 2º Drcha.A
Edificio Valrealty A
La Florida
28023 Madrid

Tel: (+34) 1 372 9814
Fax: (+34) 1 372 9645

Olicom Italy

Via Rasori 13
20145 Milano

Tel: (+39) 2 4800 3661
Fax: (+39) 2 4800 5888

Olicom Japan

4F, Shin-yokohama Daisan Toshio Bldg.
3-5-9, Shin-yokohama
Kohoku-ku
Yokohama 222-0033

Tel: (+81) 45 477 3105
Fax: (+81) 45 477 3106

Olicom Enterprise Products, Inc.

450 Donald Lynch Boulevard
Marlborough
MA 01752

Tel: (+1) 508 481 4060
Fax: (+1) 508 229 5535

Olicom Poland Sp. z o.o.

ul. Uphagena 27
80-237 Gdansk

Tel: (+48) 58 347 1451
Fax: (+48) 58 346 1238

Olicom Singapore

10 Anson Road
#15-12 International Plaza
Singapore 079903

Tel: (+65) 324 5652
Fax: (+65) 324 7019

Olicom Sweden

Kanalvägen 10 C-12, 9 vån
194 61 Upplands Väsby

Tel: (+46) 8 590 041 94
Fax: (+46) 8 590 041 96

Olicom Enterprise Products (UK) Ltd

Swan House
Peregrine Business Park
Gomm Road
High Wycombe
Bucks HP13 7DL

Tel: (+44) 1494 556 600
Fax: (+44) 1494 556 616

World Wide Web URLs

<http://www.olicom.com>

<http://www.olicom.dk>

OC-6993/1.0



* 7 1 0 0 0 1 4 9 8 *