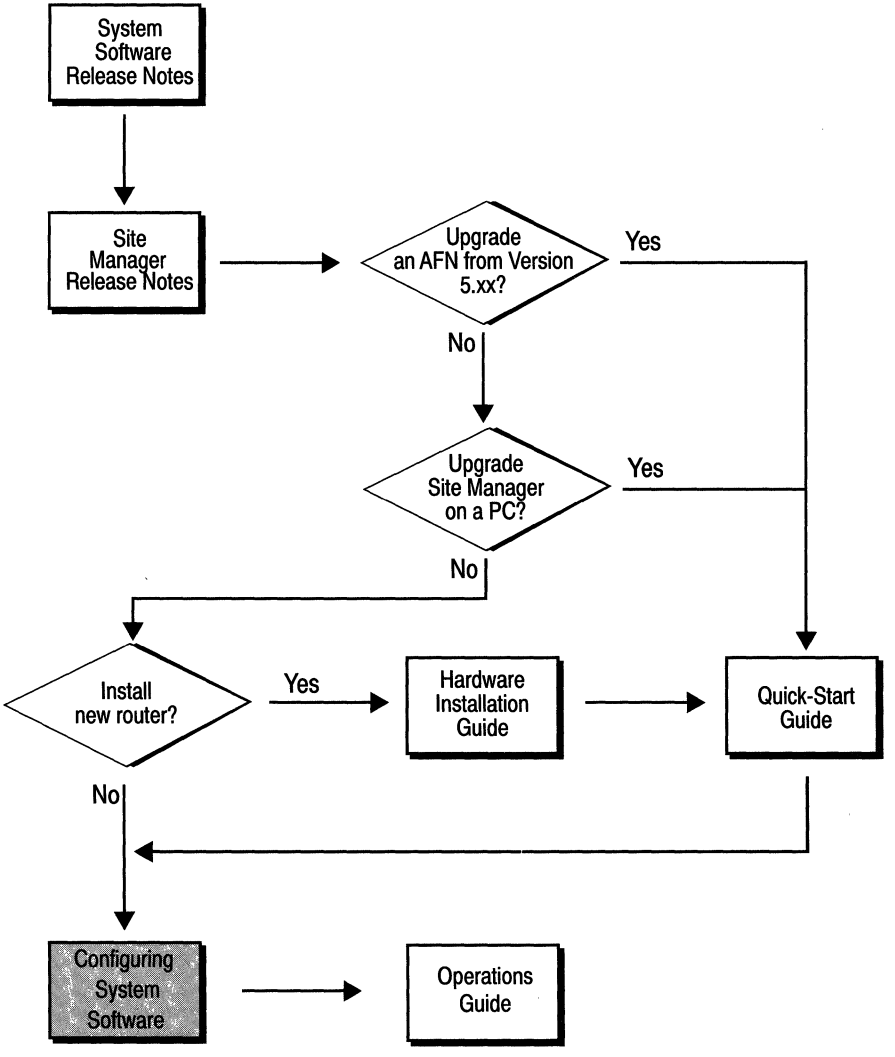
Configuring System Software
Volume II

Software Version 7.56, Site Manager Version 1.56

Reading Path



Part Number: 105544, Revision D

Wellfleet Communications, Inc., 8 Federal Street, Bedford, MA 01821

Software License

This license governs the licensing of all Wellfleet software (Software) provided to licensee for use with Wellfleet equipment (Equipment). Licensee is provided with Software in machine-readable form and related documentation. The Software provided under this license is proprietary to Wellfleet and to third parties from whom Wellfleet has acquired license rights. Wellfleet does not grant any Software license whatsoever, either explicitly or implicitly, except by acceptance of an order for either a Software license or for a Wellfleet product that is packaged with Software. Each such license is subject to the following restrictions:

1. Licensee is granted a license to use the Software when payment for the license fee is made. Upon receipt of payment, licensee is granted a personal, nontransferable, nonexclusive license to use the Software with the specific item of Equipment with which or for which it was originally acquired, including use at any of licensee's facilities to which the Equipment may be transferred, for the useful life of the Equipment unless earlier terminated by default or cancellation. Use of the Software shall be limited to such specific item of Equipment and to such facility. Software which is licensed for use on hardware not offered by Wellfleet (e.g. Site Manager) is not subject to restricted use on any Equipment, however, unless otherwise specified in the Documentation, each licensed copy of such Software may only be installed on one item of hardware at any time.
2. Licensee may use the Software with the backup Equipment only if the Equipment with which or for which it was acquired is inoperative.
3. Licensee may make a single copy of the Software (but not firmware) for safekeeping (archives) or backup purposes.
4. Licensee may modify Software (but not firmware), or combine it with other software, subject to the provision that those portions of the resulting software which incorporate licensed Software are subject to the restrictions of this license. Licensee shall not make the resulting software available for use by any third party.
5. Wellfleet and third parties from whom Wellfleet has acquired license rights shall at all times retain title to and ownership of their respective portions of the Software including new versions, new releases, updates and modifications provided to licensee. Licensee agrees and acknowledges that licensee will obtain only such rights to a license or sublicense for the Software as are specifically provided herein.

Software License (continued)

6. Licensee shall not provide, or otherwise make available, any Software, in whole or in part, in any form, to any third party. Third parties do not include consultants, subcontractors or agents of licensee who have licensee's permission to use the Software at licensee's facility, and who have agreed in writing to use the Software only in accordance with the restrictions of this license.

7. Third party owners from whom Wellfleet has acquired license rights to software that is incorporated into Wellfleet products shall have the right to enforce the provisions of this license against licensee.

8. Licensee shall not remove or obscure any copyright, patent, trademark, trade secret or similar intellectual property or restricted rights notice within or affixed to any Software and shall reproduce and affix such notice on any backup copy of Software or copies of software resulting from modification or combination performed by licensees as permitted by this license.

9. Notwithstanding any foregoing terms to the contrary, if Customer licenses the Product "Site Manager", Customer may duplicate and install the Site Manager Software as specified in the Documentation. This right is granted solely as necessary for use of the Site Manager Software on hardware installed within Customer's network. [Note: For licensees in the European Community, the Software Directive enacted by the Council of European Communities Directive dated 14 May 1991 shall apply for interoperability purposes. Licensee must notify Wellfleet in writing of any such intended examination of the Software and Wellfleet may provide review and assistance.]

10. Licensee shall not reverse assemble, reverse compile, or in any way reverse engineer the Software.

11. This license will automatically terminate upon improper handling of Software, such as by disclosure, or Wellfleet may terminate this license by written notice to licensee if licensee fails to comply with any of the material provisions of this license and fails to cure such failure within thirty (30) days after the receipt of written notice from Wellfleet. Upon termination of this license, licensee shall discontinue all use of the Software and return the Software and related documentation, including all copies, to Wellfleet.

12. Licensee's obligations under this license shall survive expiration or termination of this license.

FCC Compliance Notice: Radio Frequency Notice

The following notice regarding compliance with Federal Communications Rules pertain to the Access Feeder Node, Backbone Link Node, and Backbone Concentrator Node.

This equipment generates, uses, and can radiate radio-frequency energy. If you do not install and use this equipment according to the instruction manual, this product may interfere with radio communications. This product has been tested and found to comply with the limits for a Class A computing device, pursuant to Subpart J of Part 15 of FCC Rules; compliance with these limits provides reasonable protection against radio interference when such equipment is operated in a commercial environment. Operating this equipment in a residential area is likely to interfere with radio communications; in which case, the user, at his/her own expense, must correct the interference.

Wellfleet shielded cables must be used with this unit to ensure compliance with the Class A limits.

Canadian Department of Communications Radio Interference Regulations

This digital apparatus (the Access Feeder Node, Backbone Link Node, or Backbone Concentrator Node) does not exceed the Class A limits for radio-noise emissions from digital apparatus as set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le présent appareil numérique (le Access Feeder Node, le Backbone Link Node, et le Backbone Concentrator Node) n'émet pas de bruits radioélectriques dépassant les limites applicables au appareils numériques de Classe A prescrites dans Le Réglement sur Le Brouillage Radioélectrique Édité par Le Ministère des Communications du Canada.

SITE MANAGER SOFTWARE

SITE MANAGER SOFTWARE IS AVAILABLE FOR INSTALLATION ON EITHER SUN SPARCSTATIONS OR DOS-BASED PERSONAL COMPUTERS (PCs). SITE MANAGER MAY BE INSTALLED ON AN UNLIMITED NUMBER OF CUSTOMER SUN SPARCSTATIONS. HOWEVER, SITE MANAGER FOR DOS PCs INCLUDES DISTINCT CORPORATION'S IP RUNTIME SOFTWARE WHICH CAN BE COPIED AND INSTALLED ON UP TO 15 PCs PER NETWORK IN CONJUNCTION WITH WELLFLEET SITE MANAGER FOR DOS PCs.

Table of Contents

Volume 1

Chapter 3

Configuring Circuits

Chapter 4

Configuring Frame Relay

Chapter 5

Configuring SMDS

Chapter 6

Configuring AppleTalk

Chapter 7

Configuring the Bridge

Chapter 8

Configuring Source Routing

Chapter 9

Configuring DECnet Phase IV

Chapter 10

Configuring IP

Chapter 11

Configuring OSPF

Volume II

Chapter 12

Configuring IPX

Chapter 13

Configuring SNMP

Chapter 14

Configuring VINES

Chapter 15

Configuring XNS

Chapter 16

Configuring Filters

Chapter 17

Configuring Protocol Prioritization

Chapter 18

Booting the Wellfleet Router with the Config File

Appndix A

Site Manager Default Settings

Appendix B

IEEE Assigned Codes

Appendix C

Converting Existing Traffic Filters

Chapter 12

Configuring IPX

List of Figures

List of Tables

Configuring IPX

About this Chapter

This chapter describes how to configure the IPX router. The section *IPX Overview* identifies the services provided by the IPX router. The section *Role of the IPX Router in a Client-Server Connection* briefly describes how IPX routers provide access to servers on an IPX internetwork. The *Implementation Notes* section provides guidelines you should follow if you are configuring IPX without RIP, or IPX on a Token Ring or SMDS interface. The *Editing IPX Parameters* section describes how to use the Configuration Manager to edit the IPX parameters.

Overview

The following sections provide a brief description of IPX routing and a description of the internetworking services pertinent to the Wellfleet IPX router.

IPX Routing

The Internet Packet Exchange (IPX) Protocol is the Novell, Inc. adaptation of XNS. Wellfleet's implementation of IPX is based on Novell's *IPX Router Specification*, Version 1.10, Part No. #107-000029-001 (Novell, Inc., November 17, 1992). Like XNS, IPX has the following characteristics:

- IPX is a connectionless datagram protocol. In other words, it does *not* need a channel established for delivery.
- IPX is unreliable. Higher level protocols assume the responsibility for reliability.
- IPX uses the XNS packet format.

IPX uses different upper-layer protocols in the ISO reference model than does XNS. Those pertinent to IPX routing are described in the section *Support for Upper Layer Services*.

Note: Wellfleet's implementation of IPX supports only Directed Broadcasts of Type Packet Exchange Packet (PEP) that are destined for networks other than the one the packet originated on. The router drops Directed Broadcasts of any other type and increments the field *wfIpxInterfaceInUnknownProtos* in the IPX Interface record on the interface that the router received the packet.

According Novell's IPX Router Specification (specified above), all packets which have the IPX Packet Type set to NetBIOS Type 20 will be treated as NetBIOS Broadcast Packets (refer to the *NetBIOS Static Routing* section). The specification requires NetBIOS applications to adhere to the defined format of a NetBIOS type 20 broadcast packet.

Support for Lower Layer Services

The Wellfleet IPX router supports the following physical and data link layer protocols:

- Ethernet: 802.3 (Novell), Ethernet II, LSAP (802.5), SNAP
- Token Ring: LSAP
- FDDI: LSAP
- Frame Relay: Frame Relay SNAP
- SMDS: SMDS SNAP
- Point-to-Point (Wellfleet proprietary): Ethernet

Provision of Network Layer Services

The Wellfleet IPX router provides the following network layer support:

- Dynamic routing of IPX packets
- Static routing to other networks
- Static routing to adjacent hosts

Dynamic routing occurs automatically once you configure IPX with RIP on an interface. The sections that follow describe alternatives to dynamic routing.

Static Route Support

Static routes are manually configured routes that specify the next hop in the transmission path a datagram must follow based on the datagram's destination address. A static route specifies a transmission path to another *network*.

The Wellfleet IPX router allows you to configure static routes on each logical IPX interface. For example, in Figure 12-1 the route from the interface on Wellfleet Router Host ID 1 to Network 5 is a static route.

Static route support for IPX allows you to do the following:

- Direct all IPX traffic destined to a given network to an adjacent host.
- Reduce routing traffic by disabling RIP Supply on all or a subset of attached interfaces and manually configuring static routes.
- Eliminate all dynamic routing capabilities and all RIP supply and listen activities over an IPX interface.

Note: Unlike routes learned through RIP, static routes remain in the RIP tables until you delete them.

Warning: To establish a Data Link layer connection in a Frame Relay or SMDS network, which allows the router to send packets over a static route, you must configure an adjacent host, and edit the DLCI parameter in the IPX Adjacent Host Parameters Window.

Refer to the section *Editing Static Route Parameters* for detailed instructions and parameter definitions.

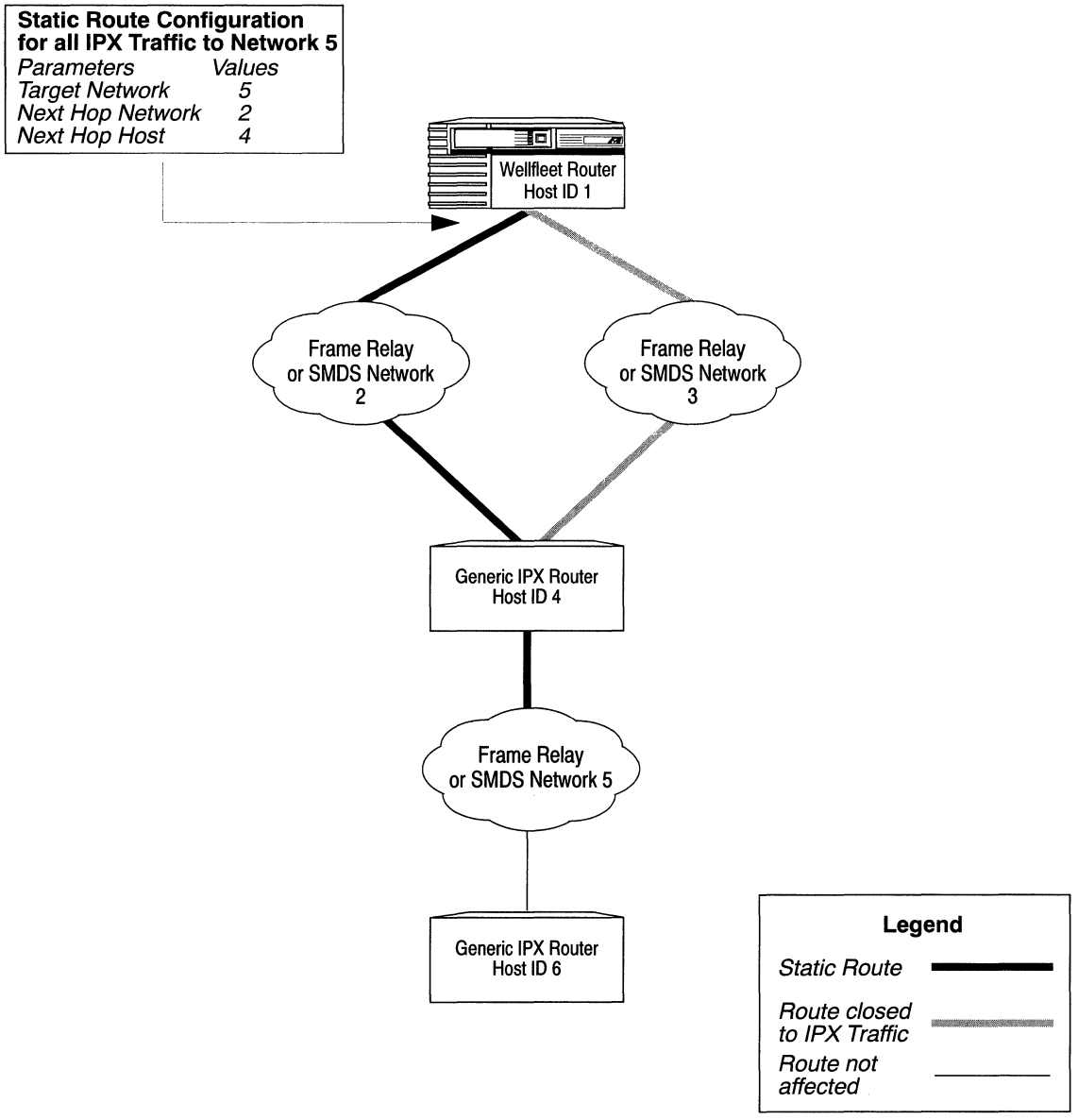


Figure 12-1. Static Route in a Sample Network

Adjacent Host Support

An adjacent host is a *network device* (that may or may not be a router) that is local to a directly connected network. For example, host 4 in Figure 12-2 is an adjacent host to Wellfleet Router Host ID 1. Host 6 is *not* an adjacent host because it is *not* connected logically to a directly adjacent network.

The Wellfleet IPX router allows you to specify static transmission paths to adjacent hosts. A static transmission path to an adjacent host establishes the data link connection necessary for packet transmission along a static route in a Frame Relay or SMDS network when RIP is not enabled. For example, in Figure 12-2 the IPX interface on Wellfleet Router Host ID 1 has host 4 configured as a statically adjacent host. This provides a data link connection that allows the static routing to occur between Host ID 1 and Network 5 in Figure 12-1.

With adjacent host support, you can do the following:

- You can configure the IPX router to map IPX addresses of network devices that are local to adjacent WANs to their associated WAN addresses.
- You can configure many static routes that use a single adjacent host as their next hop node, thereby reducing manual configuration tasks.

Note: You must use the DLCI (Data Link Control Identifier) parameter to identify a virtual circuit when you configure a static adjacent host in a Frame Relay network. You display this parameter by adding the adjacent host and then clicking the Edit button. Refer to the section *Editing Adjacent Host Parameters* for detailed instructions and parameter definitions.

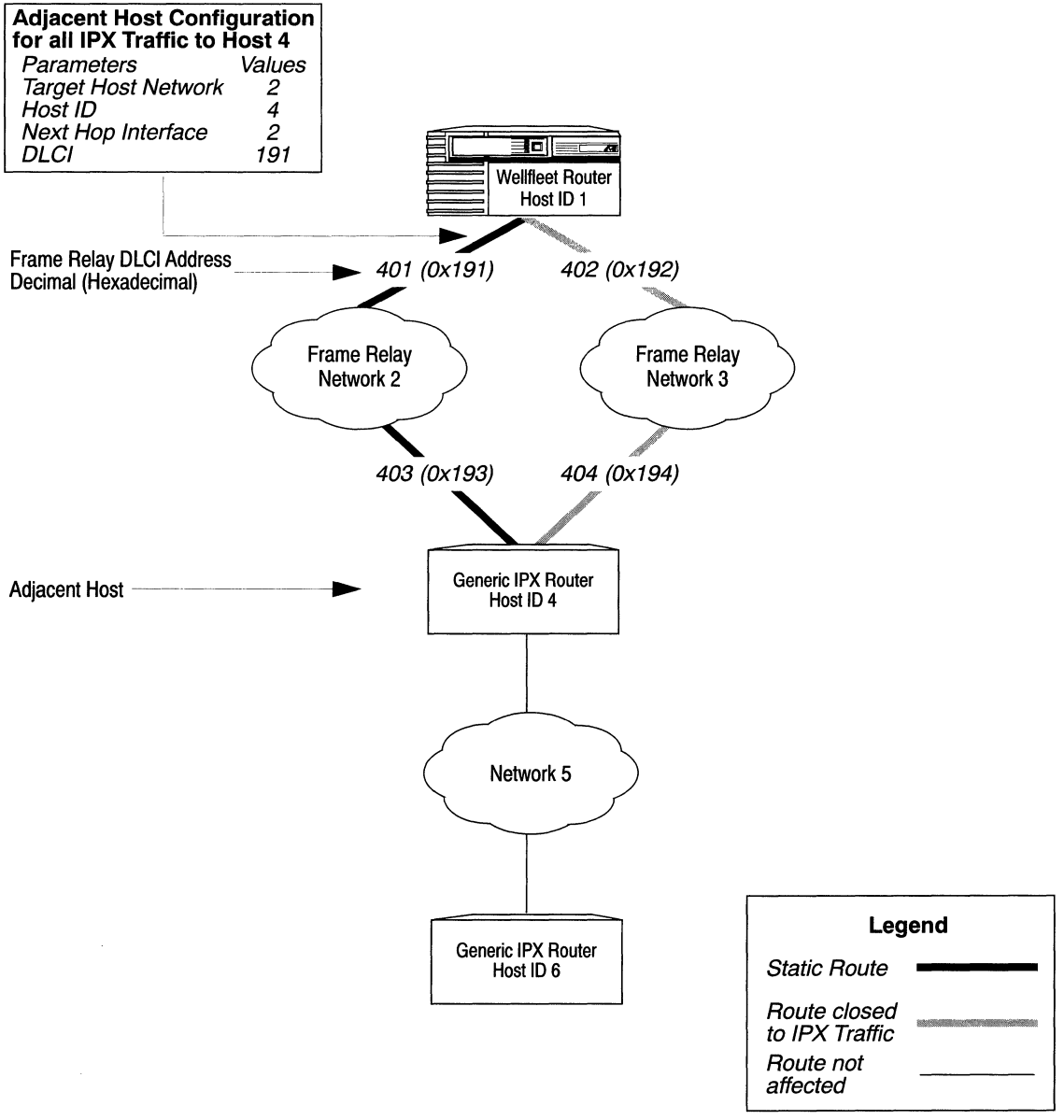


Figure 12-2. Static Adjacent Host in a Sample Network

Support for Upper Layer Services

The packet structures associated with Novell's upper-layer protocols are encapsulated within the IPX data area. The packet type and the source and destination socket number assignments designate the packet structure's protocol. The protocols pertinent to IPX routing are as follows:

- Novell's Service Advertising Protocol (SAP) provides a means for servers to advertise their services to routers and other servers.
- Novell's implementation of Routing Information Protocol (RIP) provides workstations and routers with a means for exchanging information dynamically to establish the route with the fewest hops and the minimum delay to each network.
- NetBIOS (Network Basic Input Output System) "All nets broadcast" packets.

The sections that follow describe the support the IPX Wellfleet router offers that pertain to these services.

Service Advertising Protocol

Service Advertising Protocol (SAP) enables servers to inform clients of their presence. A server makes itself known to clients by name, type, and IPX address by using the identification broadcasting services provided by SAP.

Servers broadcast service advertising packets at 60-second intervals. The packets identify the server by name, type, and network address. An IPX address identifies the server's network, host, and socket.

IPX routers maintain a database (called a bindery) containing server-specific information: name, type, IPX address, hop count, the interface to the server, and a timer value to age table entries. When an entry's timer value reaches four minutes without being refreshed, the router deletes the entry.

Each time a router receives a SAP packet, it compares the packet's contents to the contents of the bindery. If the table already contains information about the server, the router simply refreshes the age timer. If the table does *not* contain information about the server, the router adds a new entry to its table and triggers an advertisement of the new service to all connected networks. Also, each IPX router issues regularly scheduled advertisements of its table; these advertisements, issued at 60-second intervals, propagate binderies throughout the IPX network.

Clients use the IPX broadcast facility to request information about network servers. Client information requests are nearest service queries seeking information on the closest service of a specified type.

Every IPX server and IPX router on the internetwork learns about all of the other IPX servers and IPX routers through the propagation of binderies. These binderies can become very large in large internetworks. You may want to create SAP filters to control the size of these binderies, reduce bandwidth, or provide security.

You create a SAP filter by configuring an IPX interface to prevent or to allow access to servers. A SAP filter determines whether the IPX router advertises a particular service in its SAP broadcasts and responses to client requests. SAP filters affect only outgoing SAP

advertisements; the IPX router updates its own bindery according to incoming SAP packet data regardless of the status of its filters.

You can configure SAP filters using the following levels:

- You can filter service information pertaining to individual servers by editing SAP server level filters.

  At the server level, the filter matches a pattern (consisting of a *target server name* and a server type) in the bindery. The filter's Action parameter determines the action (advertise or suppress).

- You can filter service information pertaining to entire networks by editing SAP network level filters.

  At the network level, the filter matches a pattern (consisting of a *target network number* and a server type) in the bindery. The filter's Action parameter determines the action (advertise or suppress).

Each interface supports up to 50 server level and 50 network level SAP filters. The IPX router compares each filter to each pattern in the bindery to determine the contents of an advertisement.

The first pattern match that occurs determines whether the server information is advertised. The IPX router compares SAP filters to each pattern in the following order of precedence:

1. Server level filters with specific server types
2. Server level filters with wildcard server types (configured as FFFF)
3. Network level filters with specific server types and specific network numbers
4. Network level filters with wildcard server types and specific network numbers
5. Network level filters with wildcard network numbers (FFFFFFFF)

The IPX router includes the information about a service in the SAP advertisement if one of the following is true:

- The router finds a match and the filter Action is Advertise.
- The router does *not* find a match.

The IPX router excludes information about a service from an advertisement only if it matches a filter to a pattern and the Action is Suppress.

You can use wildcards to advertise or suppress all server types, all server types in a specified network, or a specific server type in all networks. Also, the order of precedence allows you to use wildcards to advertise or suppress all except for those you configure to do the opposite. For example, you can configure the IPX router to advertise only server types 4 and 5 by creating the following filters:

- A server level filter to advertise server 4
- A server level filter to advertise server 5
- A server level filter to suppress all servers

Servers 4 and 5 are advertised if they appear in the bindery because their associated filters come first in the order of precedence.

Note: The order in which you create SAP filters does *not* affect filter precedence.

Routing Information Protocol

Routing Information Protocol (RIP) provides workstations and routers with a means of exchanging information dynamically to establish the route with the fewest hops and shortest delay to each network.

Each IPX router maintains a RIP table. The RIP table contains the following information about every network in the IPX network topology:

- The network address of each network.
- The number of hops (cost) to that network.
- The number of ticks to that network. A tick is equal to about 1/18 of a second.
- The address of the next hop node to which packets destined for that network will be forwarded.

In a Wellfleet IPX router, the best path to a destination is the one with the fewest hops. The router maintains tick counts in its IPX RIP tables for use by IPX routers that use ticks to determine the best path. The use of ticks to determine the best path is what differentiates IPX RIP from XNS RIP and IP RIP.

Routers maintain RIP tables by exchanging request and response packets. Routers update their RIP tables with information from incoming response packets.

The header of each packet indicates the packet operation: request or response.

RIP request packets contain the number of the destination network in the header. A RIP request packet may be one of the following types:

- A general request broadcasted by a router to determine the fastest route to all networks on an internetwork. The value *FFFFFFFF* in the network number field within the RIP data indicates that the packet is a general request.
- A specific request broadcasted by a workstation or router to determine the fastest route to a particular network. One or more network numbers in the network number field within the RIP data indicates that the packet is a specific request.

Routers at the destination network issue RIP response packets. RIP response packets contain the network number and the number of hops and ticks required to get to the network. A RIP response may be one of the following types:

- A response to a request.
- An informational broadcast from a router issued every 60 seconds.
- An informational broadcast when a change occurs in the routing table. Examples of changes in the routing table are changes in cost information, changes to routes, aging of routes, and additions of routes to networks new to the table.
- An informational broadcast when an interface performs an orderly shutdown procedure or initializes.

To limit traffic, RIP broadcasts are limited to a router's immediate segments and are *not* forwarded by receiving routers.

Warning: The IPX router learns WAN addresses from RIP and SAP broadcasts received over WANs. The router stores IPX address/WAN address pairs for future use as next hop destinations. If RIP is not configured for a WAN interface, you must configure adjacent hosts for all transmission paths to nodes adjacent to Frame Relay or SMDS circuits when you configure an IPX interface. You must then configure static routes from the adjacent hosts to the next hop routers.

The IPX router allows you to enable RIP listen and RIP Supply functions for each IPX and/or XNS interface. When the Listen function is enabled, the IPX router adds routes received in RIP updates from neighboring routers to its own internal routing table. When the Supply function is enabled, the IPX router transmits RIP updates to routers in neighboring networks.

Configurable Split Horizon

The Split Horizon algorithm is part of Novell's specification for IPX. The Split Horizon excludes RIPs and SAPs learned from a neighbor when forwarding RIP and SAP updates to that neighbor. Its purpose is to prevent circular routes and reduce network traffic.

Split Horizon is enabled by default for each interface, in accordance with Novell's specification.

Warning: Wellfleet advises you *not* to disable Split Horizon unless it is absolutely necessary.

If you have a star or non-fully meshed Frame Relay or SMDS topology, it may be necessary to disable Split Horizon on certain interfaces in order for the routers to learn about the other networks.

A fully meshed network is a WAN in which all nodes have a logically direct connection to each other. Figure 12-3 shows a sample fully meshed network with Split Horizon enabled.

A non-fully meshed network is a WAN in which one or more nodes do not have logically direct connections to all other nodes. Figure 12-4 shows a sample non-fully meshed network with Split Horizon disabled.

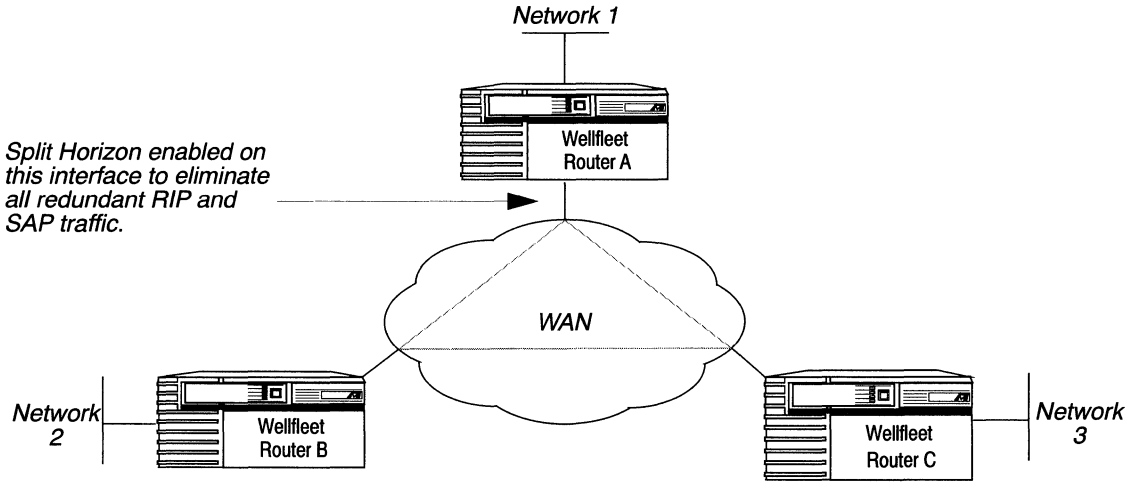


Figure 12-3. Split Horizon Enabled in a Fully Meshed Network

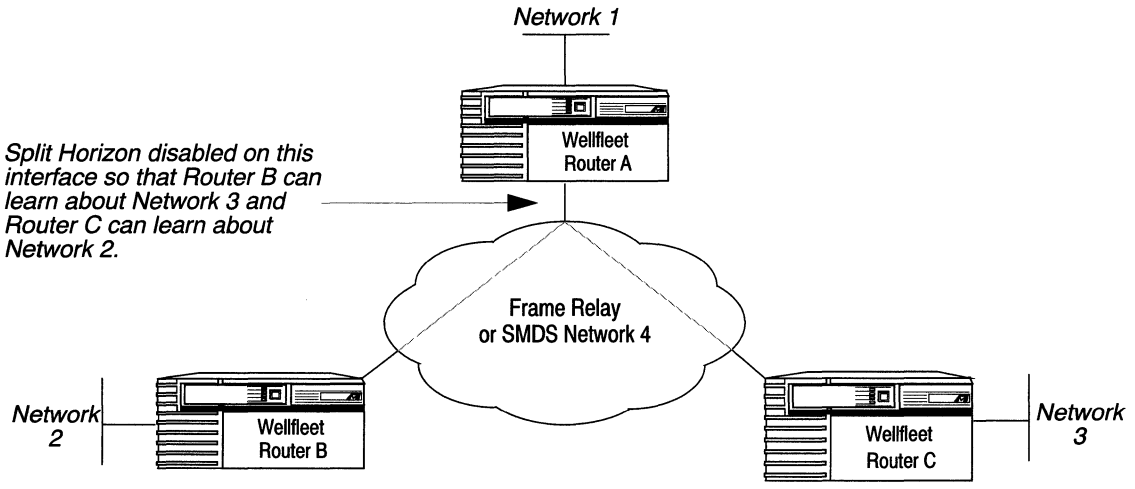


Figure 12-4. Split Horizon Disabled in a Non-Fully Meshed Network

NetBIOS Static Routing

NetBIOS was developed by Sytek for the IBM PC network. It establishes sessions (logical connections) and allows for communication between PCs.

The Wellfleet IPX router provides NetBIOS static routing to reduce network traffic. This feature is a Wellfleet enhancement to IPX routing standards.

Warning: IPX NetBIOS static routing is a nonstandard Wellfleet feature that may not interoperate with non-Wellfleet routers. A NetBIOS static route converts a NetBios broadcast packet to a NetBIOS directed broadband packet, thereby eliminating the loop checking and path tracing that is normally done on NetBIOS broadcast packets. This may cause problems with certain applications that rely on this information.

The Wellfleet IPX router allows you to configure a static route to a server name and type. After you specify the server name and type, the IPX router converts standard NetBIOS broadcast packets (of IPX Packet Type 20) to NetBIOS directed broadband packets. NetBIOS broadcast packets are sent to all accessible host IDs on all accessible IPX networks. NetBIOS directed broadband packets are sent to all host IDs on a single IPX network.

An IPX packet type of 20 indicates that the packets are NetBIOS broadcast packets. When you configure a NetBIOS static route, the IPX router inserts the target network number in the network number field of the NetBIOS packets. Refer to the section *Editing NetBIOS Static Route Parameters* for instructions on how to add a NetBIOS Static Route.

Each IPX router interface supports up to 50 NetBIOS static routes. Each NetBIOS static route specifies a NetBIOS resource name and a destination network (where the resource resides).

When NetBIOS static routes are configured on an interface, the IPX router compares all IPX NetBIOS broadcast packets received on the interface with interface-specific NetBIOS static routes. If the NetBIOS destination name found in the packet matches a table entry, the NetBIOS packet is routed to the associated destination network; if no match is found, the IPX router treats the packet as specified by the NetBIOS Accept and NetBIOS Deliver parameters.

The (NetBIOS) Accept and Deliver parameters allow you to configure each interface to accept and forward NetBIOS broadcasts. The default setting for both of these parameters is Enabled.

Note: The description that follows assumes the NetBIOS destination name found in the packet does not match an entry in the NetBIOS Static Routing table.

With Accept enabled on an interface, the IPX router accepts NetBIOS broadcast packets received on that interface. For example, in Figure 12-5 the IPX router accepts only NetBIOS broadband packets received on Interfaces 1 and 2, because the Accept parameter for those interfaces is set to Enabled.

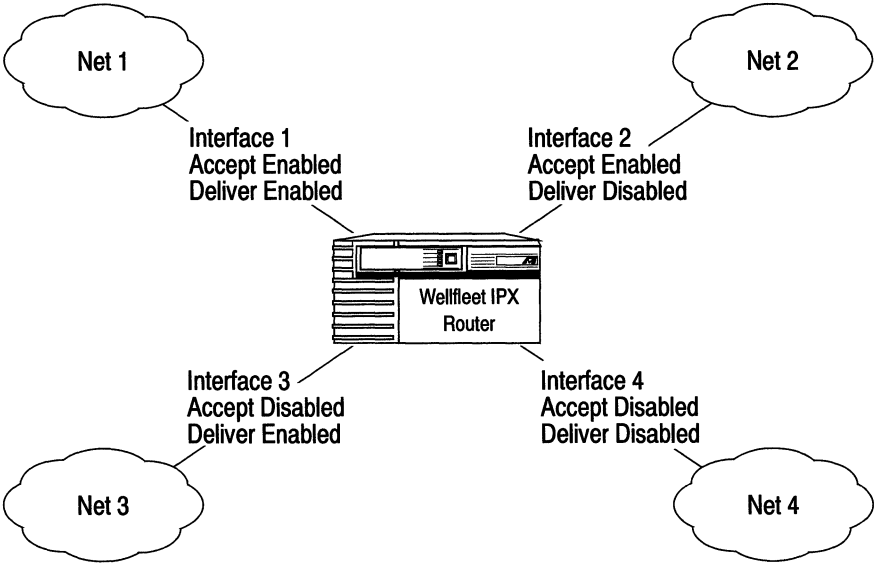


Figure 12-5. NetBIOS Directed Broadband Packets in a Sample Network

With Deliver enabled on an interface, the IPX router delivers NetBIOS "all networks broadcast" packets that are routed to that interface. For example, in Figure 12-5 the IPX router delivers only NetBIOS "all network broadcast" packets to Interfaces 1 and 3, because the Deliver parameter for those interfaces is set to Enabled.

The Accept parameter of the interface receiving NetBIOS "all networks broadcast" packets and the Deliver parameter of the other interface must both be set to Enabled for delivery of such packets to occur. For example, Interface 1 can deliver only packets from Interface 2 to Net 1 because Interface 2 is the only other interface whose Accept parameter is set to Enabled.

Thus, NetBIOS client applications on Network 1 can initiate and establish sessions with NetBIOS server applications only on Network 3. NetBIOS client applications on Network 2 can initiate and establish sessions with NetBIOS server applications only on Networks 1 and 3. Client applications on Networks 3 and 4 cannot initiate any sessions with NetBIOS server applications via the IPX router.

Refer to the section *Editing IPX Interface Parameters* for instructions on how to disable the Accept and Deliver parameters.

Source Route End Node Support

The IPX router allows you to configure source route end node support for Token Ring networks on each interface. Configuring source route end node support enables the coexistence of bridging and routing in the same IBM source route bridging environment. With end node support enabled, end stations that support both source route bridging and IPX can use source routing to traverse bridged networks.

In a source routing network, every end station supplies each frame it sends out with the necessary route descriptors so that it can be source routed across the network. Thus, in order for IPX routers to route packets across a source routing network, *they must act like end stations*, supplying route descriptors within each packet before they send it onto the network.

With end node support enabled, the Wellfleet IPX router does the following whenever it receives a packet and determines the packet's next hop is located across a source routing network:

- Adds the necessary RIF information to the packet's MAC header.
- Sends the packet to the network where it is source routed toward the next hop.

After the peer router receives the packet from the Token Ring network, it strips off the RIF field and continues to route the packet toward the destination network address (see Figure 12-6).

You configure source route end node support on a per-interface basis by setting the TR End Station parameter to Enable. See the section *Editing IPX Interface Parameters* for instructions on enabling this parameter.

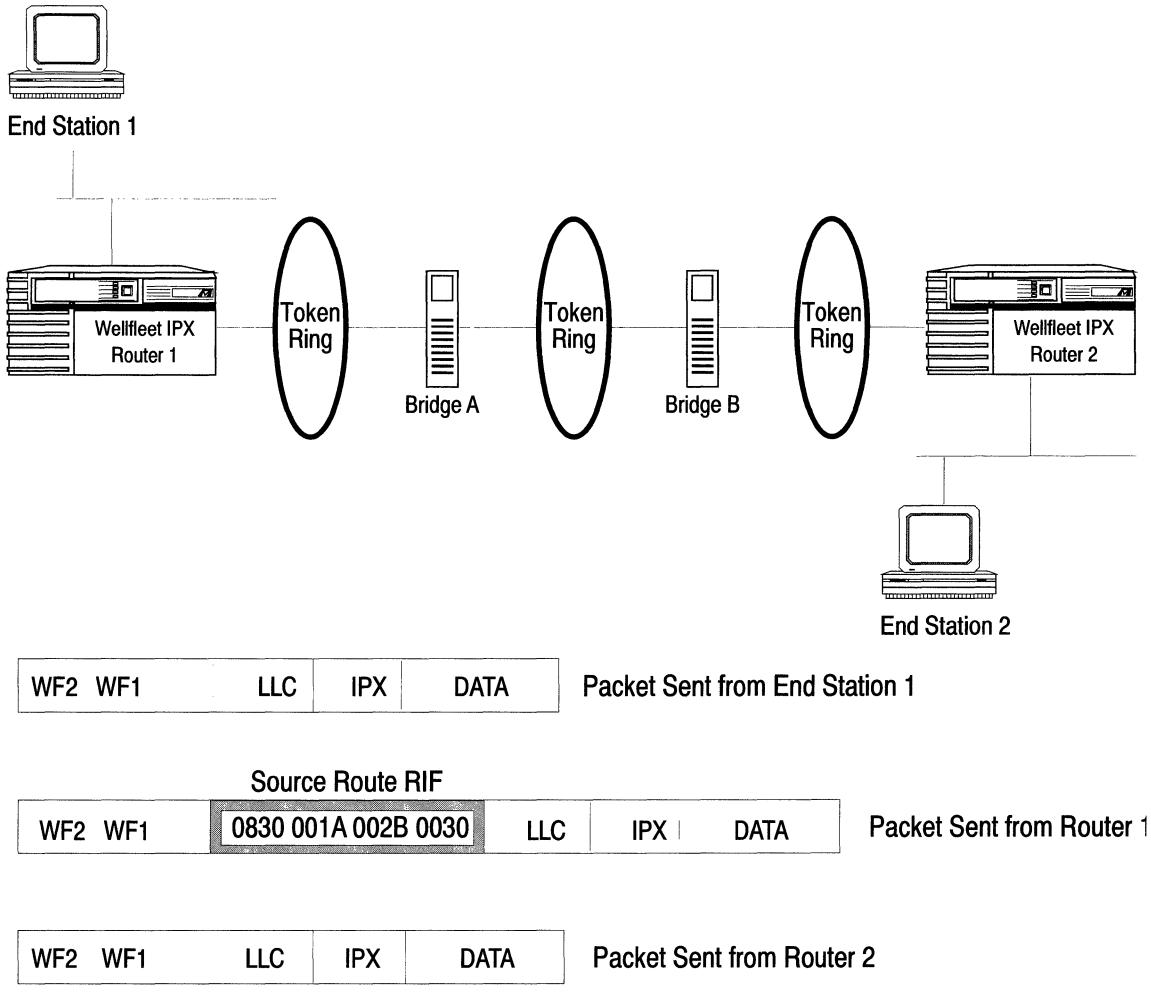


Figure 12-6. IPX Routers Source Routing Across a Token Ring Network

Role of the IPX Router in a Client-Server Connection

This section describes how IPX routers provide access to servers on an IPX internetwork.

The IPX internetwork maintains binderies and RIP tables as described in the sections *Service Advertising Protocol* and *Routing Information Protocol* earlier in this chapter.

IPX routers provide access to servers on an IPX internetwork as follows:

1. Regularly scheduled SAP broadcasts from all file servers advertise their services to the local IPX routers. Each router maintains a bindery to be used whenever a SAP request is received. If regular SAP broadcasts from a file server stop, the local router ages the entry out from its bindery.
2. A client sends a get_nearest_service SAP request whenever it needs a service type.

   For example, in Figure 12-7 client A sends a get_nearest_ service SAP request of type 4. (Type 4 is a file service.)
3. If the service resides on the same network, the associated server receives the request and responds. The local router does not respond because its bindery indicates the service is available on the client's network. In this case, client-router communications stop until the client sends the next get_nearest_service SAP request.

   If the service does *not* reside on the same network, the router responds because its bindery indicates the service is *not* available on the client's network. The SAP response contains the network and host IPX address of the nearest device offering the service.

   In cases where multiple servers are available and those servers are the same number of hops away, the Wellfleet router selects the server whose name is the lowest.

   In this case, Wellfleet router host 2 responds because its bindery indicates that the service is *not* available on the client's

network. The SAP response includes the IPX host address (3) and network address (6) of the nearest device offering the service.

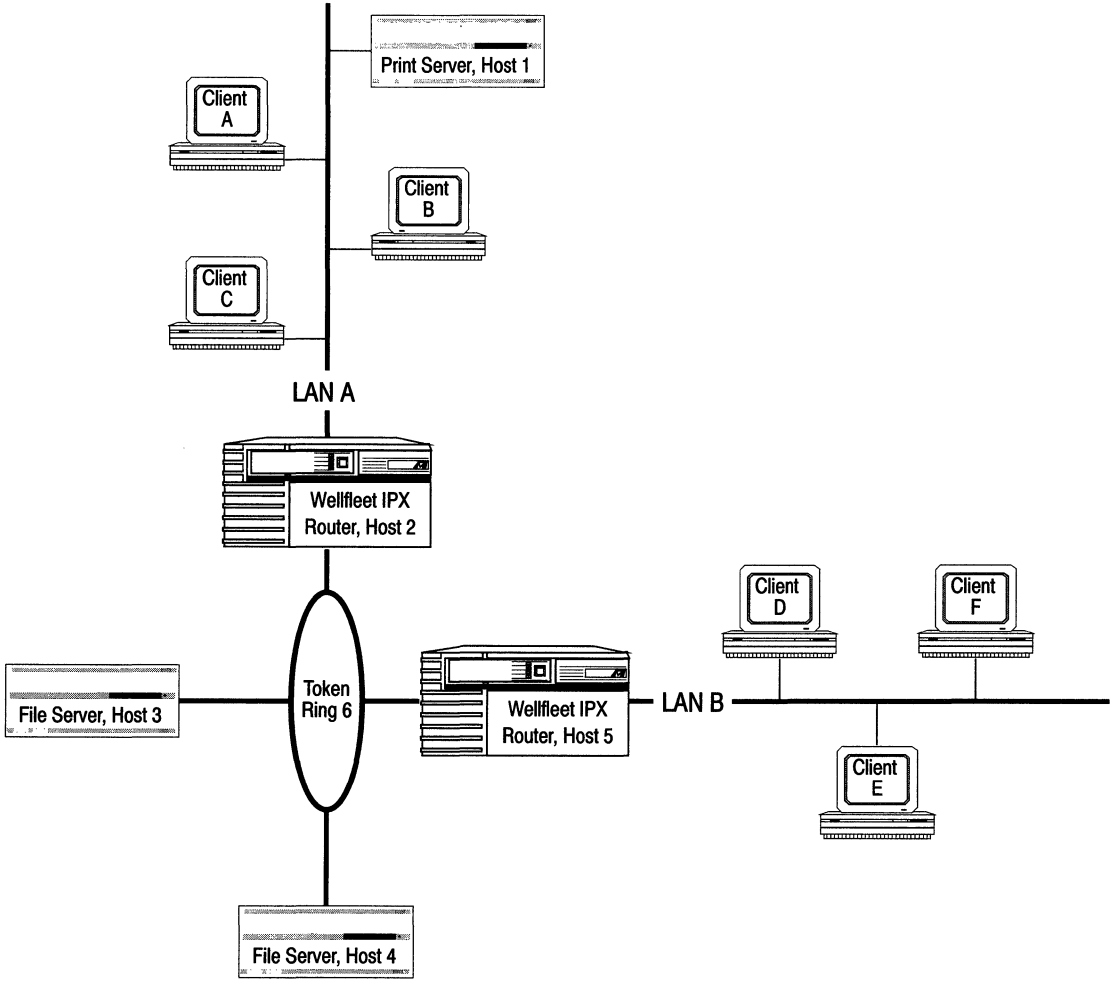


Figure 12-7. Sample IPX Network

4. The client then forwards a RIP request packet to the router. This packet requests the best path to the host that was learned from the SAP response.

   In this case, client A forwards a RIP request to Wellfleet router host 2. The RIP request packet contains a request for the route to network 6.

5. The router on the same network as the client refers to its RIP table and sends a RIP response to the client. The RIP response identifies the network on which the desired server resides.

   In this case, Wellfleet router host 2 refers to its RIP table and sends a RIP response back to client A.

6. The client inserts the node address learned from the SAP response and the destination network address learned from the RIP response into the headers of all subsequent IPX packets to be sent to that server.

   In this case, client A inserts IPX host ID 3 and destination network address 6 into the headers of all subsequent IPX packets.

7. The client forwards the IPX packets to the IPX router.

   In this case, client A forwards the IPX packets to Wellfleet router host 2.

8. The IPX router forwards the packets to the network identified by the destination network address.

   In this case, Wellfleet router host 2 forwards the packets to network 6.

IPX Bibliography

The following documents provided technical detail on IPX protocol implementation.

Advanced Netware, V2.0 Internet Packet Exchange Protocol (IPX) with Asynchronous Event Scheduler (Novell, Inc., March 19, 1986)

IPX Router Specification, Version 1.10, Part No. #107-000029-001 (Novell, Inc., November 17, 1992)

Implementation Notes

You should refer to the sections that follow only if you are configuring one of the following:

- IPX without RIP
- IPX on a Token Ring interface

Otherwise, refer to the section *Editing IPX Parameters*.

Configuring IPX without RIP

The IPX router learns WAN addresses from RIP and SAP broadcasts received over WANs. The router stores the IPX address/WAN address pairs in its RIP Table for future determination of next hop destinations.

Every IPX router on the internetwork learns about all of the other IPX routers through the propagation of RIP Tables. These tables can become very large in large internetworks. You may want to configure IPX without RIP to control the size of these tables and reduce bandwidth. However, you must do the following when you configure an IPX WAN interface without RIP:

1. Configure an adjacent host, and edit the DLCI parameter in the IPX Adjacent Host Parameters Window for each host on an adjacent Frame Relay or SMDS network.

   Refer to the section *Editing Adjacent Host Parameters* for detailed instructions.

2. Configure a static route to the next hop router for each adjacent host.

   Refer to the section *Editing Static Route Parameters* for detailed instructions.

Configuring a MAC Address on a Token Ring Interface

Any physical interface that can run in promiscuous mode, such as LANCE, ILACC, and FSI, allows multiple protocols to register a MAC address for which the protocol software can listen. Therefore, IPX can register its host number as the MAC address for each interface. However, if IPX is running over a Token Ring interface, you must override the default Token Ring MAC address and configure the same MAC address for both Token Ring and IPX, using one of the following methods:

- Leave the Host Number parameter blank if the box-generated host number does not conflict with another host number on a directly connected network. Refer to the section *Editing IPX Global Parameters* for detailed instructions. Then ensure that the MAC Address Select parameter of *every* Token Ring interface on which IPX is running is set to its default setting (Boxwide) as follows:

    a. Select the Circuits/Edit Circuits option from the Configuration Manager Window.

    b. Select the Token Ring interface in the Circuit List Window and click the Edit button.

    c. Select the Lines option in the Circuit Definition Window.

    d. Select the interface from the Edit Lines Window and click the Edit button.

    e. Ensure the MAC Address Select parameter is set to its default setting (Boxwide) in the Token Ring Parameters Window.

    f. Repeat steps b through e for every Token Ring interface on which IPX is running.

    Note: Refer to the chapter *Configuring Circuits* for more information about configuring circuits.

- Enter a MAC address in the Host Number parameter only if the box-generated host number conflicts with another host number on a directly connected network. Refer to the section *Editing IPX Global Parameters* for detailed instructions. Then set the MAC Address Select parameter of *every* Token Ring interface on which IPX is running to Cnfg as follows:

    a. Select the Circuits/Edit Circuits option from the Configuration Manager Window.

    b. Select the Token Ring circuit in the Circuit List Window and click the Edit button.

    c. Select the Lines option in the Circuit Definition Window.

    d. Select the interface from the Edit Lines Window and click the Edit button.

    e. Set the MAC Address Select parameter to Cnfg in the Token Ring Parameters Window.

    f. Repeat steps b through e for every Token Ring circuit on which IPX is running.

    Note: Refer to the chapter *Configuring Circuits* for more information about configuring circuits.

Editing IPX Parameters

As you configure the IPX router, you supply information that it uses to route packets through an IPX Internet. The instructions in the following sections describe how to edit IPX global and interface parameters. This section assumes you have configured an interface to support IPX. Refer to the chapter *Configuring Circuits* for instructions.

You use the Configuration Manager to edit IPX parameters. The configuration function you wish to perform determines the type of parameters you edit. Table 12-1 lists each configuration function and the corresponding section in this chapter.

Table 12-1. IPX Parameters and Configuration Functions

To Do the Following:	See this Section:
Enable or Disable IPX on the entire Wellfleet router.	*Editing IPX Global Parameters*
Reconfigure IPX on an interface.	*Editing IPX Interface Parameters*
Reconfigure the Routing Information Protocol (RIP) on an interface.	*Editing RIP Interface Parameters*
Add, edit, and delete adjacent hosts.	*Editing Adjacent Host Parameters*
Add, edit, and delete static routes.	*Editing Static Route Parameters*
Add, edit, and delete NetBIOS static routes.	*Editing NetBIOS Static Route Parameters*
Add, edit, and delete network level SAP filters.	*Editing Network Level Sap Filters*
Add, edit, and delete network level SAP filters.	*Editing Server Level Sap Filters*
Delete IPX from the entire Wellfleet router.	*Deleting IPX from the Wellfleet Router*

The sections that follow describe how to access and edit IPX parameters. The following information is provided for each parameter:

- Wellfleet default
- Valid options
- Parameter's function
- Instructions for setting the parameter

To edit the IPX parameters, you begin from the Wellfleet Configuration Manager Window, the first window displayed when you enter the Configuration Manager application (see Figure 12-8). Select the Protocols/IPX option. The IPX configuration options are displayed.

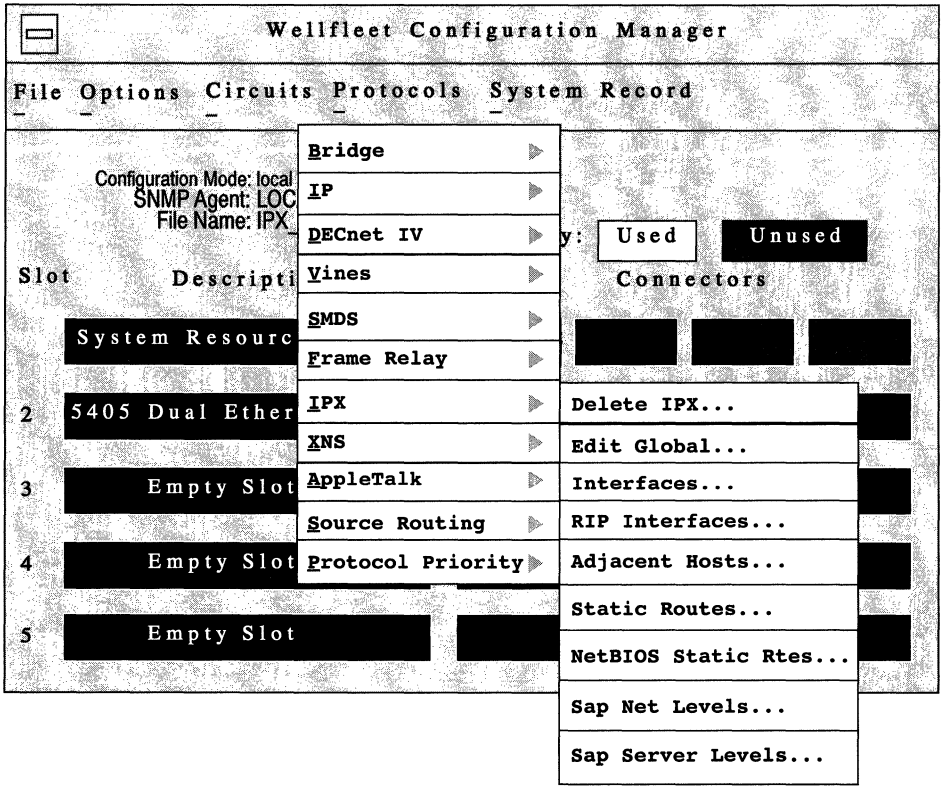


Figure 12-8. Wellfleet Configuration Manager Window

Editing IPX Global Parameters

To edit IPX Global parameters, begin at the Wellfleet Configuration Manager Window and proceed as follows:

1. Select the Protocols/IPX/Edit Global option.

   The Edit IPX Global Parameters Window appears (see Figure 12-9).

2. Edit those parameters you wish to change.
3. Click the Save button to save your changes and exit the window.

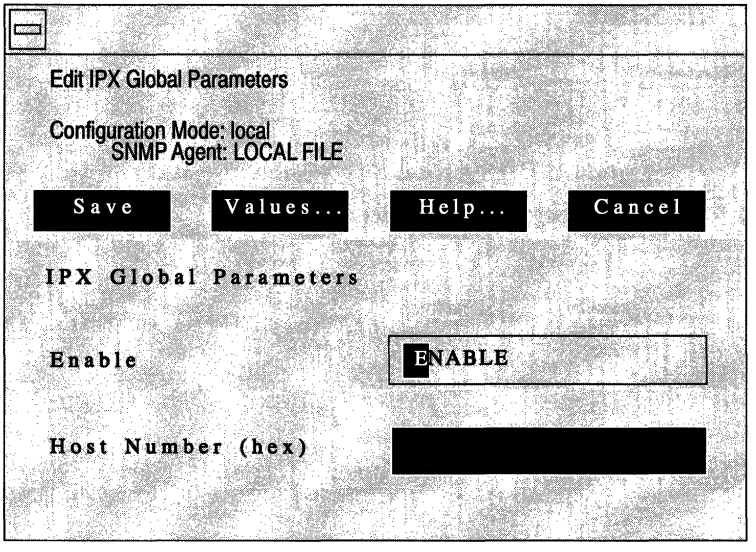


Figure 12-9. Edit IPX Global Parameters Window

A description of the parameters in this window follows.

Parameter:	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this parameter to Enable when you add IPX support to an interface.
Options:	Enable/Disable
Function:	Enables or disables IPX on the entire Wellfleet router.
Instructions:	Select Enable if you have previously disabled the IPX router software and now wish to enable it. Select Disable to disable the IPX router software.

Parameter:	**Host Number**
Wellfleet Default:	The Configuration Manager automatically generates a unique 6-byte host number from the Wellfleet router's serial number if you do not enter a value. (The automatically generated number is not displayed.)
Options:	Any host number
Function:	Sets the host ID and the source MAC address of the box.
Instructions:	Do not enter a number in this box if you want the Configuration Manager to generate a host number automatically or if the interface is on a Token Ring circuit and you are setting the Token Ring Mac Address Select parameter to Boxwide. Enter the MAC address in hexadecimal notation only if the interface is on a Token Ring circuit and you are setting the Token Ring MAC Address Select parameter to Cnfg.

Refer to the section *Configuring a MAC Address on a Token Ring Interface* for more information about this parameter.

Editing IPX Interface Parameters

When you added IPX to an interface, it took the IPX default settings. You can change these default settings by editing the IPX interface parameters.

To edit IPX interface parameters, begin at the Wellfleet Configuration Manager Window and proceed as follows:

1. Select the Protocols/IPX/Interfaces option to display the IPX Interfaces Window (see Figure 12-10).

   This window displays the network address in hexadecimal format of each interface you named when you added a circuit.

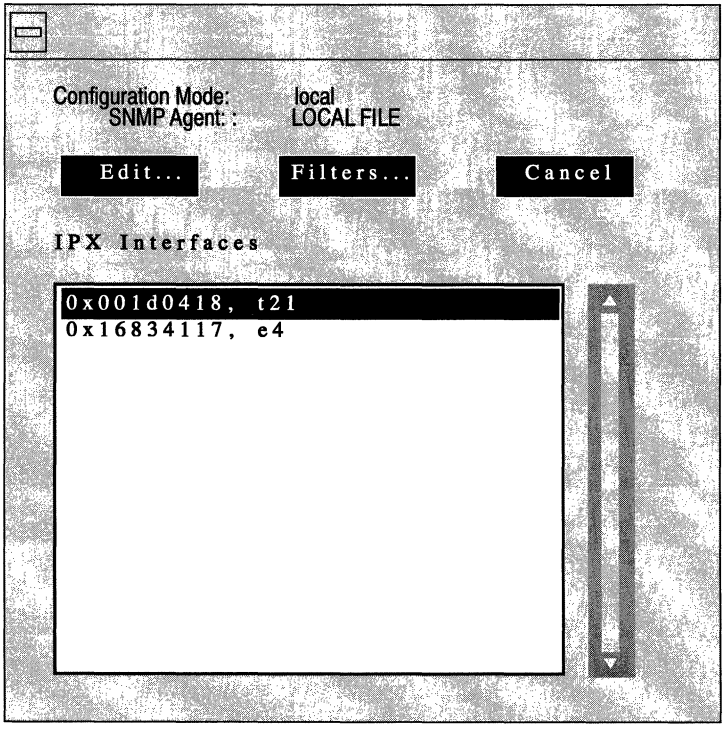


Figure 12-10. IPX Interfaces Window

2. Select the interface you wish to edit.
3. Click the Edit button to display the IPX Interface Parameters Window for that interface (see Figure 12-11).
4. Edit those parameters you wish to change.
5. Click the Save button to save your changes and exit the window.

Note: When you reconfigure an interface in dynamic mode, IPX restarts on that interface.

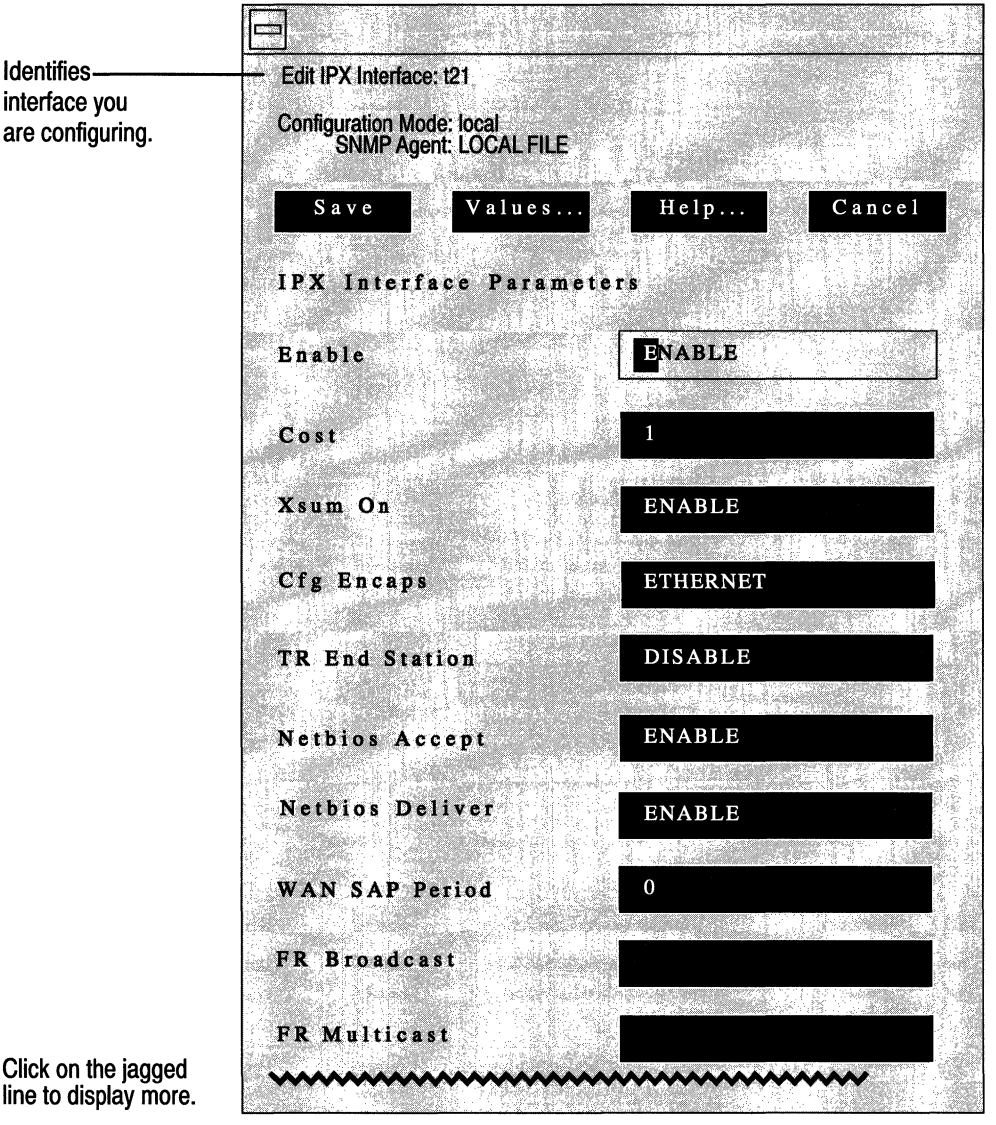


Figure 12-11. IPX Interface Parameters Window

A description of the parameters in this window follows.

Warning: When you reconfigure an interface in dynamic mode and select the Save button, IPX restarts on that interface.

Parameter:	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this interface-specific parameter to Enable when you add IPX support to this interface.
Options:	Enable/Disable
Function:	Enables or disables IPX routing on this interface.
Instructions:	Select Enable if you previously set this parameter to Disable and now wish the interface to support IPX routing. Select Disable only if you wish to disable IPX routing over this interface.

Parameter:	**Cost**
Wellfleet Default:	1 (for each hop)
Options:	0 to 15
Function:	Sets the cost (number of hops) for this interface. This parameter allows you to configure the shortest path. The cost is added to routes learned on this interface through RIP and is specified in subsequent RIP packets sent to other interfaces. IPX disposes of the packet when its hop count surpasses 15.
Instructions:	Enter the interface cost value. Standard RIP implementation assigns a cost of 1. Increasing this value causes the upper bound of 15 set by the RIP Network Diameter to be attained more rapidly.

Parameter :	**Xsum on**
Wellfleet Default:	On
Options:	On/Off
Function:	Novell IPX does not implement checksumming and writes *FFFFFF* in the checksum field regardless of the setting of this parameter.
Instructions:	The setting of this parameter does *not* affect IPX checksumming.

Parameter :	**Cfg Encaps**
Wellfleet Default:	Ethernet
Options:	Ethernet/LSAP/Novell/SNAP
Function:	Specifies supported encapsulation methods for supported media.
Instructions:	Select the encapsulation method you wish to use. Ensure the encapsulation method matches that of the servers on the same LAN. The media types and encapsulation methods they support are as follows:

Ethernet supports Ethernet, LSAP, Novell, and SNAP.

FDDI supports LSAP, and SNAP.

Frame Relay supports Frame Relay SNAP.

SMDS supports SMDS SNAP.

Synchronous supports Ethernet. (If you select any other options on a Synchronous circuit, the software overrides the selection and uses Ethernet.

Token Ring supports LSAP.

Wellfleet Point-to-Point supports Ethernet.

Parameter :	**TR End Station**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Enables or disables source routing on this interface.
Instructions:	Select Enable if this interface connects to a Token Ring network over which packets will be bridged. Select Disable only if you wish to disable source routing over this interface.

Parameter :	**NetBIOS Accept**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Enables or disables acceptance of all NetBIOS broadcast packets originating from the interface.
Instructions:	Select Enable if you previously set this parameter to Disable and now wish the interface to accept NetBIOS broadcast packets received over this interface. Select Disable only if you wish to disable acceptance of NetBIOS broadcast packets received over this interface.

Note: Refer to the *NetBIOS Static Routing* section of this chapter for a discussion of the relationship between the NetBIOS Accept and the NetBIOS Deliver parameters.

Parameter :	**NetBIOS Deliver**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Enables or disables deliverance of all NetBIOS broadcast packets that were routed to the interface.
Instructions:	Select Enable if you previously set this parameter to Disable and now wish the interface to forward NetBIOS broadcast packets that were routed to the interface. Select Disable only if you wish the interface to drop NetBIOS broadcast packets that were routed to the interface.

Parameter:	**WAN SAP Period**
Wellfleet Default:	1
Options:	1 to 99 (minutes)
Function:	Specifies the interval at which the IPX router transmits periodic SAP advertisements called General Server Responses (GSRs). This parameter does not affect SAP advertisements generated in response to bindery changes or client requests.
Instructions:	Use the standard interval, 1 minute, if the interface is connected to a LAN. Enter another value from 2 to 99 to decrease the frequency of GSR transmissions if the interface is connected to a WAN and you want to decrease traffic. The standard IPX advertisement is 1. Enter 0 if you want to disable periodic GSR transmission. You should disable GSR transmission with great care; the loss of a single SAP advertisement can result in unsynchronized binderies at both ends of the link.

Warning: This parameter is Wellfleet-compatible only. Do not change the default setting in mixed router environments. When this parameter is set to anything other than the default value of 1, ensure that the Wellfleet router at the other end of the point-to-point link is configured with the identical value.

Parameter:	**FR Broadcast**
Wellfleet Default:	ffffff (not displayed)
Options:	Default value or a user-specified Frame Relay broadcast address.
Function:	Specifies a Frame Relay broadcast address for this IPX interface.
Instructions:	Leave blank to accept the default value. With the default value, the IPX router sends all broadcast traffic through all logical connections associated with the IPX interface you are configuring. Broadcast traffic includes RIP and SAP broadcasts. Enter a Frame Relay broadcast address to send all broadcast traffic through the IPX interface you are configuring.

Parameter:	**FR Multicast**
Wellfleet Default:	ffffff (not displayed)
Options:	Default value or a user-specified Frame Relay multicast address.
Function:	Specifies a Frame Relay multicast address for this IPX interface.
Instructions:	Leave blank to accept the default value. With the default value, the IPX router sends all multicast traffic through all logical connections associated with the IPX interface you are configuring. Enter a Frame Relay multicast address to send all multicast traffic through the IPX interface you are configuring.

Parameter :	**Split Horizon**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Excludes RIPs and SAPs learned from a neighbor when forwarding RIP and SAP updates to that neighbor.
Instructions:	Select Enable if you previously set this parameter to Disable and now wish the router *not* to transmit RIP and SAP updates received from the interface over the same interface. Select Disable only if you wish to transmit RIP and SAP updates received from the interface over the same interface.

Warning: Wellfleet advises you *not* to disable Split Horizon unless it is absolutely necessary.

Refer to the section *Configurable Split Horizons* for more information about this parameter.

Editing RIP Interface Parameters

Once you enable RIP on an interface, you can edit that interface in the RIP Interface Parameters Window for that interface. You enable RIP when you add a circuit. For instructions on how to enable RIP on an interface, see the *Configuring Circuits* chapter.

To edit RIP interface parameters for an IPX interface, begin at the Wellfleet Configuration Manager Window and proceed as follows:

1. Select the Protocols/IPX/RIP Interfaces option.

   The IPX RIP Interfaces Window appears (see Figure 12-12). This window displays the network address of each interface you named when you added a circuit.

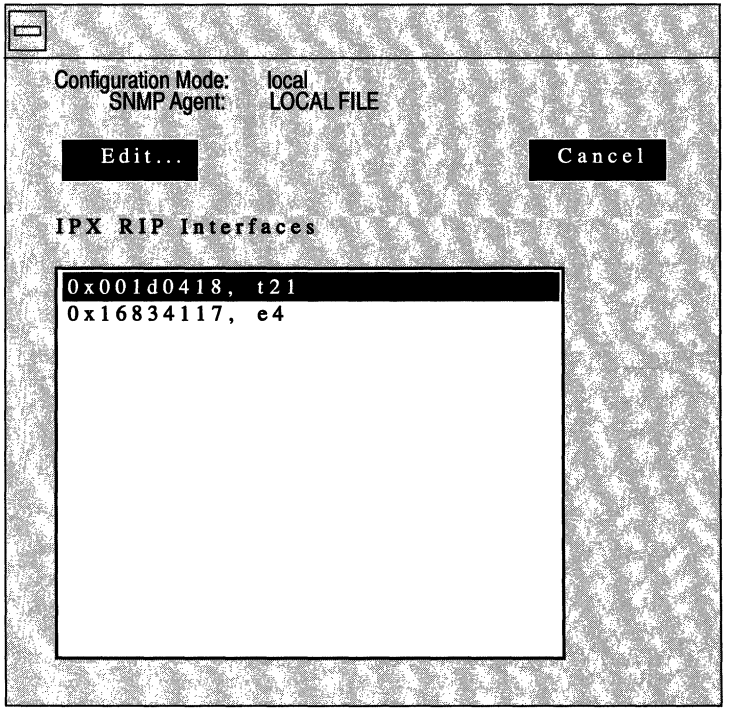


Figure 12-12. IPX RIP Interfaces Window

2. Select the interface you wish to edit.
3. Click on the Edit button.

   The RIP Interface Parameters Window appears (see Figure 12-13).
4. Edit those parameters you wish to change.
5. Click the Save button to save your changes and exit the window.

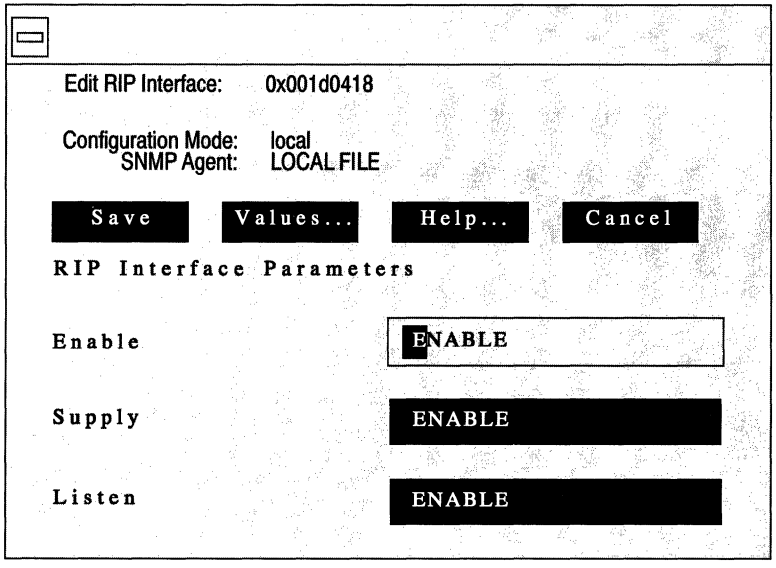


Figure 12-13. RIP Interface Parameters Window

A description of the parameters in this window follows.

Parameter:	**Enable**
Wellfleet Default:	If you enabled RIP when you added the circuit or if you edited this circuit to support RIP, the Configuration Manager automatically sets this interface-specific RIP Enable parameter to Enable; otherwise, it is set to Disable.
Options:	Enable/Disable
Function:	Specifies whether the Routing Information Protocol (RIP) is enabled on this interface.
Instructions:	Select Enable to enable RIP on this interface. Select Disable to disable RIP on this interface.

Parameter:	**Supply**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether the interface transmits all RIP updates to routers in neighboring networks.
Instructions:	Select Enable to configure the interface to transmit all RIP updates. Select Disable to prohibit the interface from transmitting all RIP updates.

Parameter :	**Listen**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether this interface listens to RIP updates from neighboring networks.
Instructions:	Select Enable to configure this interface to listen to RIP updates, and, thus, add received routing information to its internal routing table. Select Disable to configure the interface to ignore RIP updates from neighboring routers. Thus, the interface does not add received routing information to its internal routing table.

Note: If this parameter is set to Enable, a route filter can still prohibit the interface from updating its internal routing tables.

Editing Adjacent Host Parameters

The sections that follow describe how to add, edit, and delete adjacent host routes. You perform these functions from the IPX Adjacent Hosts Window (see Figure 12-14). Begin at the Wellfleet Configuration Manager Window and select the Protocols/IPX/Adjacent Hosts option. The IPX Adjacent Hosts Window appears.

Refer to the following sections to add, edit, and delete adjacent host routes.

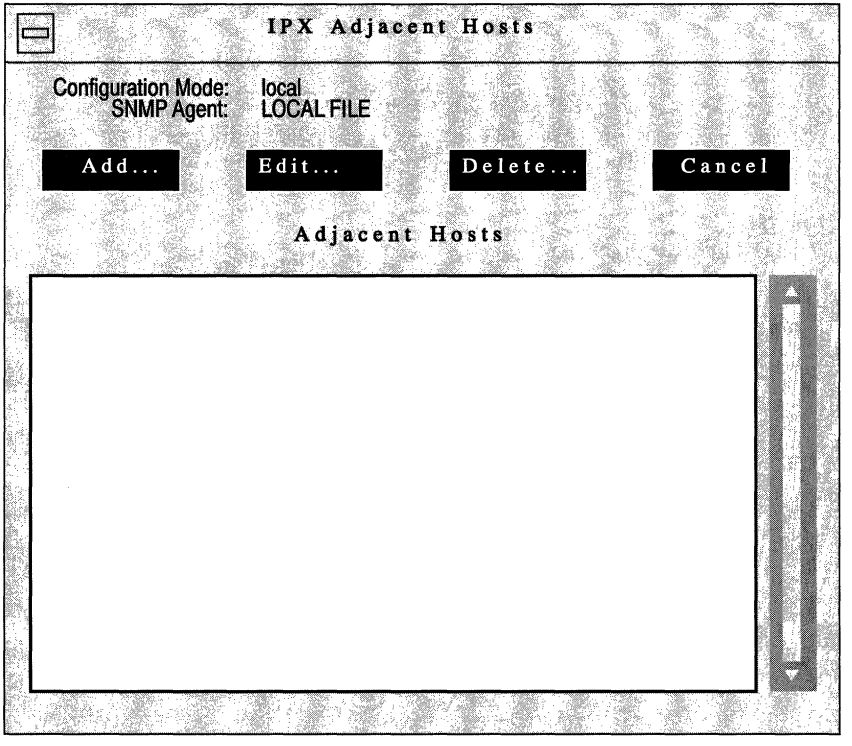


Figure 12-14. IPX Adjacent Hosts Window

Adding an Adjacent Host

To add an adjacent host, begin at the IPX Adjacent Hosts Window (see Figure 12-14) and proceed as follows:

1. Click on the Add button.

   The Add Adjacent Host Window appears (see Figure 12-15). This window contains the parameters required to add an adjacent host.

2. Edit those parameters you wish to change.

3. Click on the Add Adjacent Host button.

   The IPX Adjacent Host Parameters Window appears (see Figure 12-16). The next section describes how to edit the parameters in the IPX Adjacent Host Parameters Window.

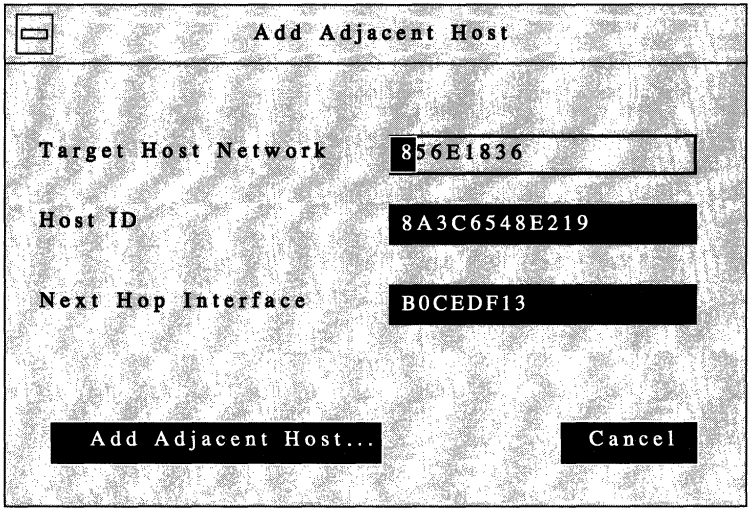


Figure 12-15. Add Adjacent Host Window

The parameters in the Add Adjacent Host Window are as follows.

Parameter :	**Target Host Network**
Wellfleet Default:	None
Options:	Valid network address of the static adjacent host.
Function:	Specifies the network address of the static adjacent host.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Host ID**
Wellfleet Default:	None
Options:	Valid host ID of the static adjacent host.
Function:	Specifies the host ID of the device for which you wish to configure an adjacent host.
Instructions:	Enter a host ID of up to 12 hexadecimal characters.

Parameter :	**Next Hop Interface**
Wellfleet Default:	None
Options:	Configured network address of the next hop.
Function:	Specifies the network address of the next hop.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Editing an Adjacent Host

You edit an adjacent host to change the settings of configurable adjacent host parameters, including the default settings.

The Configuration Manager does not allow you to change the Target Host Network and Host ID parameters. If you wish to change these parameters, you must delete the adjacent host and configure a new adjacent host. However, you can reconfigure all other parameters associated with an adjacent host.

The IPX Adjacent Host Parameters Window (see Figure 12-16) appears automatically when you follow the procedure to add an adjacent host as described in the previous section. To edit an existing adjacent host, begin at the IPX Adjacent Hosts Window (see Figure 12-14) and proceed as follows:

1. Select the adjacent host you wish to edit.
2. Click on the Edit button.

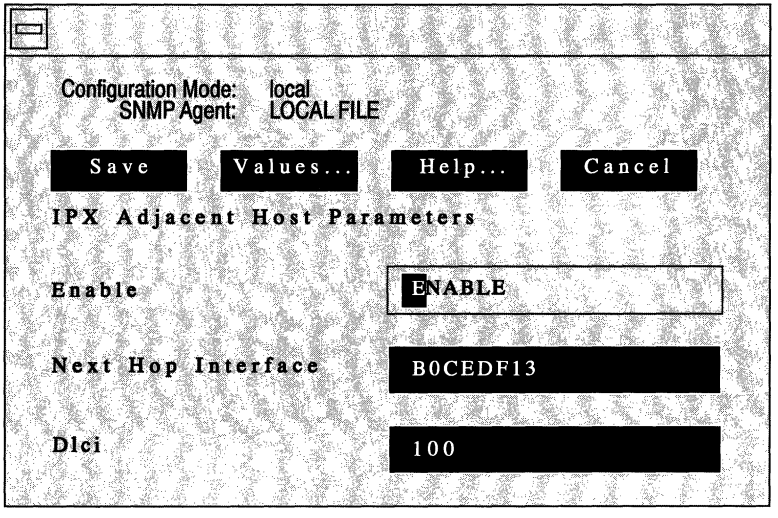


Figure 12-16. IPX Adjacent Host Parameters Window

When the IPX Adjacent Host Parameters Window is displayed, proceed as follows:

1. Edit those parameters you wish to change.
2. Click the Save button to save your changes and exit the window.

The parameters in this window are as follows.

Parameter :	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this parameter to Enable when you click on the Add Adjacent Host button in the Add Adjacent Host Window.
Options:	Enable/Disable
Function:	Specifies the state (active or inactive) of the adjacent host in the IPX routing tables.
Instructions:	Select Disable to make the adjacent host record inactive in the IPX routing table; the IPX router will not consider this adjacent host. Select Enable to make the adjacent host record active again in the IPX routing table.

Parameter :	**Next Hop Interface**
Wellfleet Default:	None
Options:	Configured network address of the next hop.
Function:	Specifies the network address of the next hop.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Dlci**
Wellfleet Default:	None
Options:	Data Link Control Identifier
Function:	Identifies the virtual circuit in a Frame Relay or SMDS network.
Instructions:	Enter a DLCI of up to 16 hexadecimal characters if the interface is on a Frame Relay or SMDS network. Leave blank if the interface is *not* on a Frame Relay or SMDS network.

Warning: The router cannot pass traffic through an interface to an adjacent host on a Frame Relay or SMDS network if the adjacent host is configured without the correct DLCI.

Deleting an Adjacent Host

To delete an adjacent host, select the adjacent host you wish to delete in the IPX Adjacent Hosts Window, and click on the Delete button (see Figure 12-14). The Delete IPX Adjacent Host Window appears. Click on the Delete button to delete the adjacent host.

Editing Static Route Parameters

IPX static routes are user-specified transmission paths that IPX internet packets follow. You configure static routes when you want to restrict the paths that packets can follow. Static routes, like routes learned through RIP, are maintained in the IPX routing table. Unlike routes learned through RIP, however, static routes do not time out. Static routes remain in the IPX routing table until they are reconfigured manually.

The sections that follow describe how to add, edit, and delete IPX static routes. You perform these functions from the IPX Static Routes Window (see Figure 12-17). Begin at the Wellfleet Configuration Manager Window and select the Protocols/IPX/Static Routes option. The IPX Static Routes Window appears.

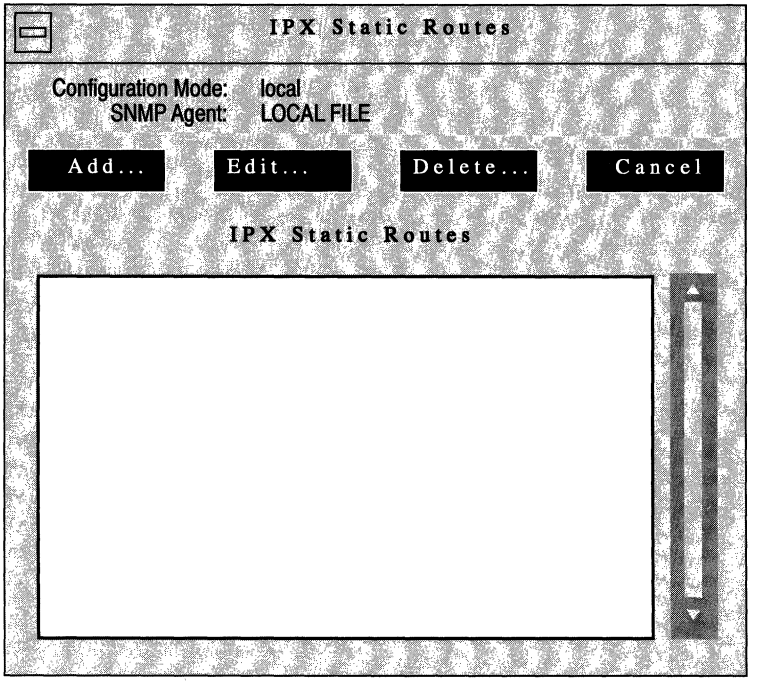


Figure 12-17. IPX Static Routes Window

Refer to the following sections to add, edit, and delete static routes.

Warning: To establish a Data Link layer connection in a Frame Relay or SMDS network, which allows the router to send packets over a static route, you must configure an adjacent host, and edit the DLCI parameter in the IPX Adjacent Host Parameters Window.

Adding a Static Route

To add a static route, begin at the IPX Static Routes Window (see Figure 12-17) and proceed as follows:

1. Click on the Add button.

   The IPX Add Static Route Window appears (see Figure 12-18). This window contains the parameters required to add a static route. (Static route parameters with default values are described in the next section.)

2. Edit those parameters you wish to change.

3. Click on the Add Static Route button.

   The IPX Static Route Parameters Window appears (see Figure 12-19). The next section describes how to edit the parameters in the IPX Static Route Parameters Window.

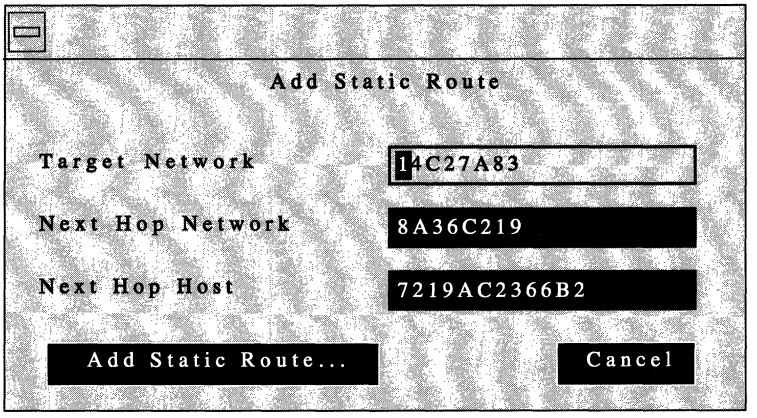


Figure 12-18. IPX Add Static Route Window

The parameters in the IPX Add Static Route Window are as follows.

Parameter :	**Target Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the address of the network to which you wish to configure the static route.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Next Hop Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the network address of the next hop.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Next Hop Host**
Wellfleet Default:	None
Options:	Any valid host address in hexadecimal notation.
Function:	Specifies the address of the host to which you wish to configure the static route.
Instructions:	Enter a host address of up to 12 hexadecimal characters.

Editing a Static Route

You edit a static route to change the settings of configurable static route parameters, including the default settings.

The Configuration Manager does not allow you to reconfigure the Target Network and Next Hop Network parameters for a static route. If you wish to change these parameters, you must delete the static route and add a new route with the proper information. However, you can reconfigure all other parameters associated with a static route.

The IPX Static Route Parameters Window (see Figure 12-19) appears automatically when you follow the procedure to add a static route as described in the previous section. To edit an existing static route, begin at the IPX Static Routes Window (see Figure 12-17) and proceed as follows:

1. Select the static route you wish to edit.
2. Click on the Edit button.

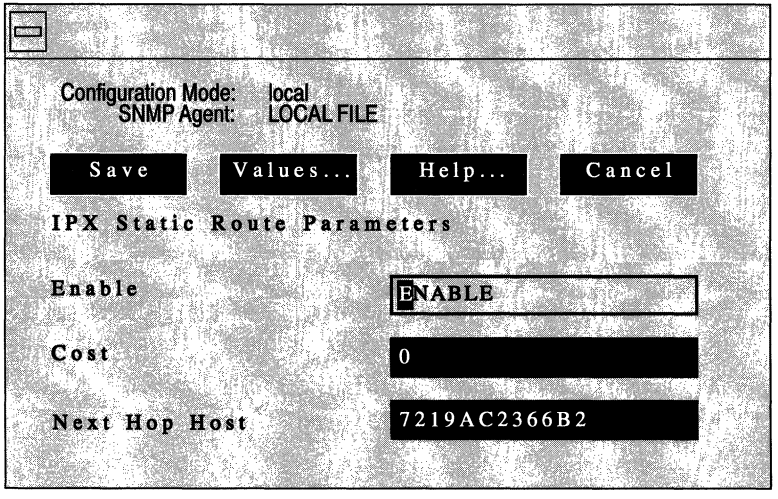


Figure 12-19. IPX Static Route Parameters Window

When the IPX Static Route Parameters Window is displayed, proceed as follows:

1. Edit those parameters you wish to change.
2. Click the Save button to save your changes and exit the window.

The parameters in the IPX Static Route Parameters window are as follows.

Parameter :	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this parameter to Enable when you click on the Add Static Route button in the Add IPX Static Route Window.
Options:	Enable/Disable
Function:	Specifies the state (active or inactive) of the static route record in the IPX routing tables.
Instructions:	Select Disable to make the static route record inactive in the IPX routing table; the IPX router will not consider this static route. Select Enable to make the static route record active in the IPX routing table.

Parameter:	**Cost**
Wellfleet Default:	0
Options:	0 to 15
Function:	Specifies the number of router hops added to an IPX data packet. The IPX router uses Cost when determining the best route for a datagram to follow. The Cost is also propagated through RIP. The default setting of 0 for static routes gives them priority over RIP-learned routes.
Instructions:	Enter the number of router hops.

Parameter:	**Next Hop Host**
Wellfleet Default:	None
Options:	Any valid host address in hexadecimal notation
Function:	Specifies the address of the host to which you wish to configure the static route.
Instructions:	Enter a host address of up to 12 hexadecimal characters.

Deleting a Static Route

To delete a static route, first select the static route you wish to delete in the IPX Static Routes Window (see Figure 12-17). Then click on the Delete button to display the Delete IPX Static Route Window. Click on the Delete button to delete the static route.

Editing NetBIOS Static Route Parameters

The Wellfleet NetBIOS Static Route function allows you to reduce NetBIOS network traffic by configuring a NetBIOS static route to a server name and type. The IPX router then restricts broadcast NetBIOS packets, which are usually forwarded to all network interfaces on a single network.

You can add, edit, and delete NetBIOS static routes to other networks, regardless of the routers used in those networks. You perform these functions from the IPX NetBIOS Static Routes Window (see Figure 12-20). Begin at the Wellfleet Configuration Manager Window and select the Protocols/IPX/NetBIOS Static Rtes option. The IPX NetBIOS Static Routes Window appears.

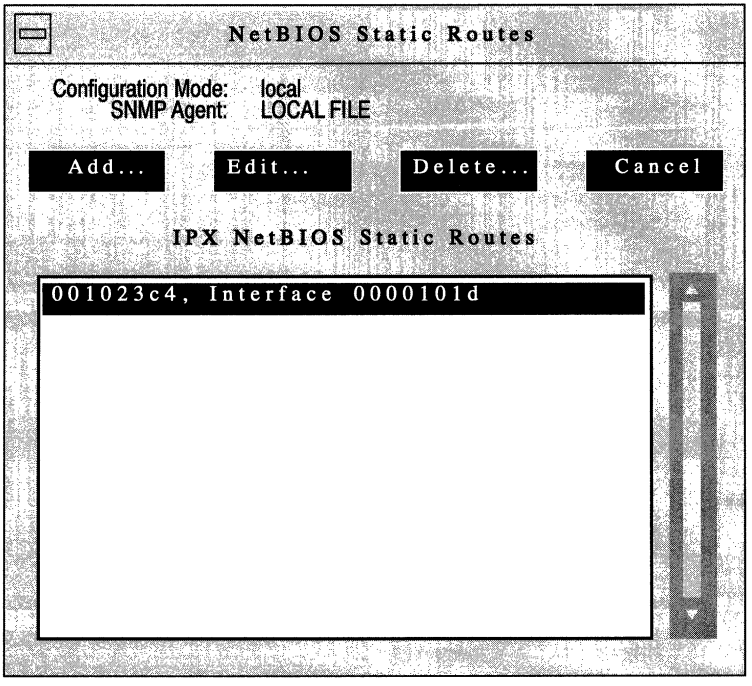


Figure 12-20. IPX NetBIOS Static Routes Window

The IPX NetBIOS Static Routes Window displays each NetBIOS Static Route you add in hexadecimal notation in the following order:

<Target Network>, Interface <Interface>

Refer to the following sections to add, edit, and delete NetBIOS Static Routes.

Adding a NetBIOS Static Route

To add a NetBIOS static route, begin at the IPX NetBIOS Static Routes Window (see Figure 12-20) and proceed as follows:

1. Click on the Add button.

   The NetBIOS Add Static Route Window appears (see Figure 12-21). This window contains the parameters required to add a NetBIOS static route. (NetBIOS static route parameters with default values are described in the next section.)

2. Edit those parameters you wish to change.
3. Click on the Add Static Route button.

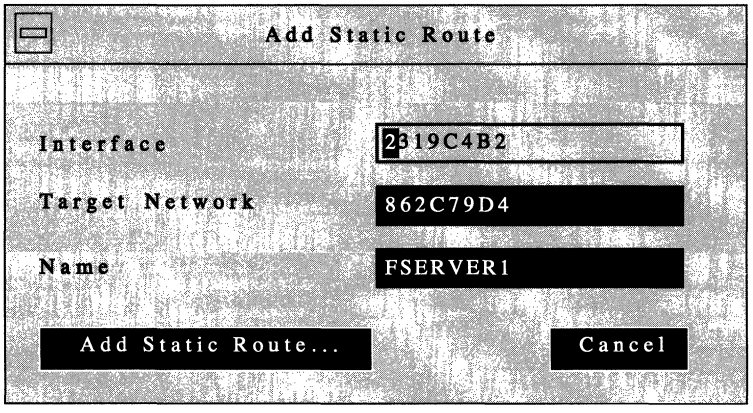


Figure 12-21. NetBIOS Add Static Route Window

The parameters in the NetBIOS Add Static Route Window are as follows.

Parameter :	**Interface**
Wellfleet Default:	None
Options:	Any configured interface address in hexadecimal notation.
Function:	Specifies the address of the interface to which you wish to configure the NetBIOS static route.
Instructions:	Enter an interface address of up to eight hexadecimal characters.

Parameter :	**Target Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the address of the destination network to which you wish to configure the NetBIOS static route.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter:	**Name**
Wellfleet Default:	None
Options:	Any valid Novell server name from 2 to 47 characters. The name cannot begin with a period, cannot contain a space, and cannot contain the following characters: " * + , / \ | : ; = < > ? [ ]
Function:	Specifies the name of the NetBIOS target.
Instructions:	Enter the name of the NetBIOS target.

Editing a NetBIOS Static Route

You edit a NetBIOS Static Route to change the settings of configurable NetBIOS Static Route parameters, including the default settings.

The Configuration Manager does not allow you to reconfigure the interface for a static route. If you wish to change these parameters, you must delete the static route and add a new route. However, you can reconfigure all other parameters associated with a static route.

The IPX NetBIOS Static Route Parameters Window (see Figure 12-22) appears automatically when you follow the procedure to add a NetBIOS static route as described in the previous section. To edit an existing NetBIOS static route, begin at the IPX NetBIOS Static Routes Window (see Figure 12-20) and proceed as follows:

1. Select the static route you wish to edit.
2. Click on the Edit button.

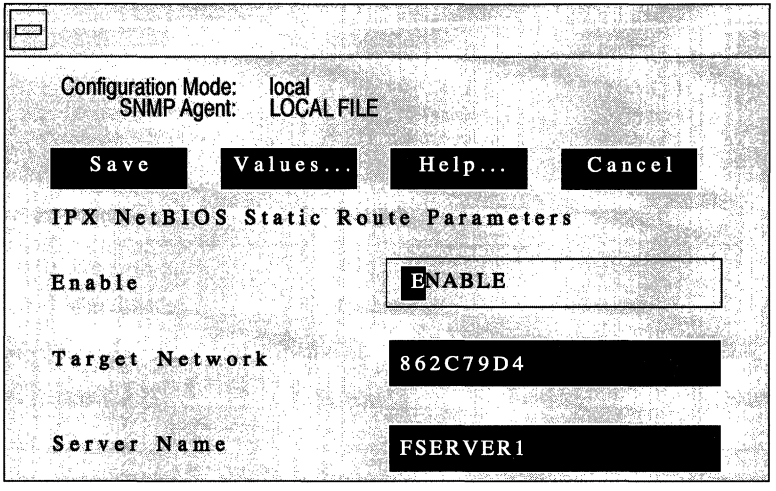


Figure 12-22. IPX NetBIOS Static Route Parameters Window

When the IPX NetBIOS Static Route Parameters Window is displayed, proceed as follows:

1. Edit those parameters you wish to change.
2. Click the Save button to save your changes and exit the window.

The parameters in this window are as follows.

Parameter :	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this parameter to Enable when you click on the Add Static Route button in the Add NetBIOS Static Route Window.
Options:	Enable/Disable
Function:	Specifies the state (active or inactive) of the static route record in the NetBIOS routing table.
Instructions:	Select Disable to make the static route record inactive in the NetBIOS routing table; the IPX router will not consider this static route. Select Enable to make the static route record active in the NetBIOS routing table.

Parameter :	**Target Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation
Function:	Specifies the address of the network to which you wish to configure the NetBIOS static route.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Server Name**
Wellfleet Default:	None
Options:	Any valid Novell server name from 2 to 47 characters. The name cannot begin with a period, cannot contain a space, and cannot contain the following characters: " * + , / \ | : ; = < > ? [ ]
Function:	Specifies the name of the NetBIOS target.
Instructions:	Enter the name of the NetBIOS target.

Deleting a NetBIOS Static Route

To delete a static route, first select the static route you wish to delete in the IPX NetBIOS Static Routes Window (see Figure 12-20). Then click on the Delete button to display the Delete NetBIOS Static Route Window. Click on the Delete button to delete the static route.

Editing Network Level SAP Filter Parameters

The Wellfleet network level SAP filters function allows you to reduce IPX SAP network traffic by configuring network level SAP filters.

You can add, edit, and delete network level SAP filters for each interface. You perform these functions from the IPX SAP Network Level Window (see Figure 12-23). Begin at the Wellfleet Configuration Manager Window and select the Protocols/IPX/Sap Net Levels option. The IPX SAP Network Level Window appears.

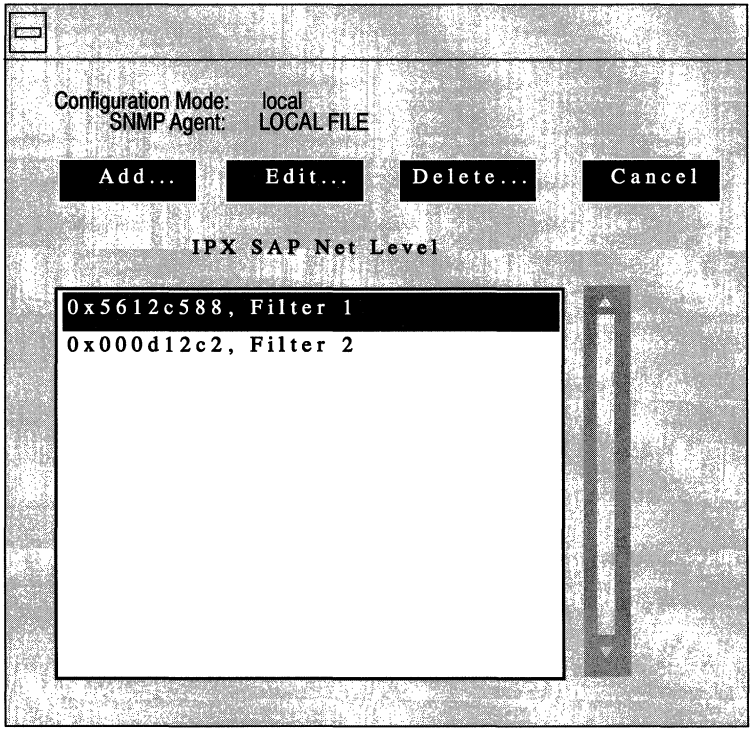


Figure 12-23. IPX SAP Network Level Window

The IPX SAP Network Level Window displays each network level SAP filter you add in hexadecimal notation in the following order:

<Interface No.>, Filter <Filter No.>

The interface and filter numbers are automatically assigned by the system. The next section describes the <Filter No.> parameter.

Adding a Network Level Sap Filter

To add a network level SAP filter, begin at the IPX SAP Network Level Window (see Figure 12-23) and proceed as follows:

1. Click on the Add button.

   The Add SAP Network Filters Window appears (see Figure 12-24). This window contains the parameters required to add a network level SAP filter. (The parameters with default values are described in the next section.)

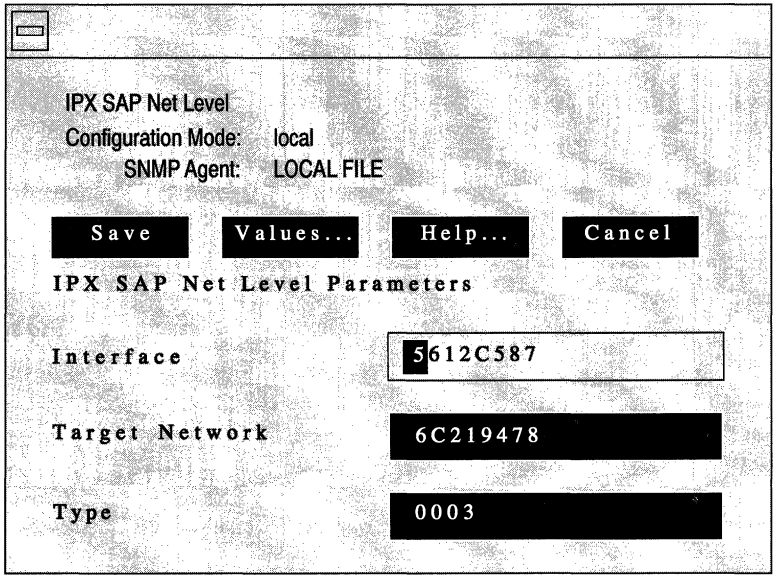


Figure 12-24. Add SAP Network Filters Window

2. Edit those parameters you wish to change.
3. Click the Save button to save your changes and exit the window.

The parameters in this window are as follows.

Parameter :	**Interface**
Wellfleet Default:	None
Options:	Any configured interface address in hexadecimal notation.
Function:	Specifies the address of the interface to which you wish to configure a network level SAP filter.
Instructions:	Enter an interface address of up to eight hexadecimal characters.

Parameter :	**Target Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the address of the network to which you wish to configure the filter.
Instructions:	Enter a network address of up to eight hexadecimal characters. You can specify all networks by entering FFFFFFFF.

Parameter: **Type**

Wellfleet Default: None

Options: Any valid Novell server type number in 4-digit hexadecimal format.

Function: Specifies the type of server to monitor.

Instructions: Enter the server type number in 4-digit hexadecimal format. Include leading zeros. Refer to Table 12-2 for a current list of the well-known server types.

Table 12-2. Well-Known Server Types

Server Type	Hexadecimal Identifier
Wild	FFFF
Unknown	0000
Print Server	0003
File Server	0004
Job Server	0005
Archive Server	0009
Remote Bridge Server	0024
Advertising Print Server	0047
Reserved Up To	8000

Editing a Network Level Sap Filter

You edit a network level SAP filter to change the settings of configurable network level SAP filter parameters, including the default settings.

To edit a network level SAP filter, begin at the IPX SAP Network Level Window (see Figure 12-23) and proceed as follows:

1. Select the filter you wish to edit.
2. Click on the Edit button.

   The IPX SAP Network Level Parameters Window appears (see Figure 12-25).

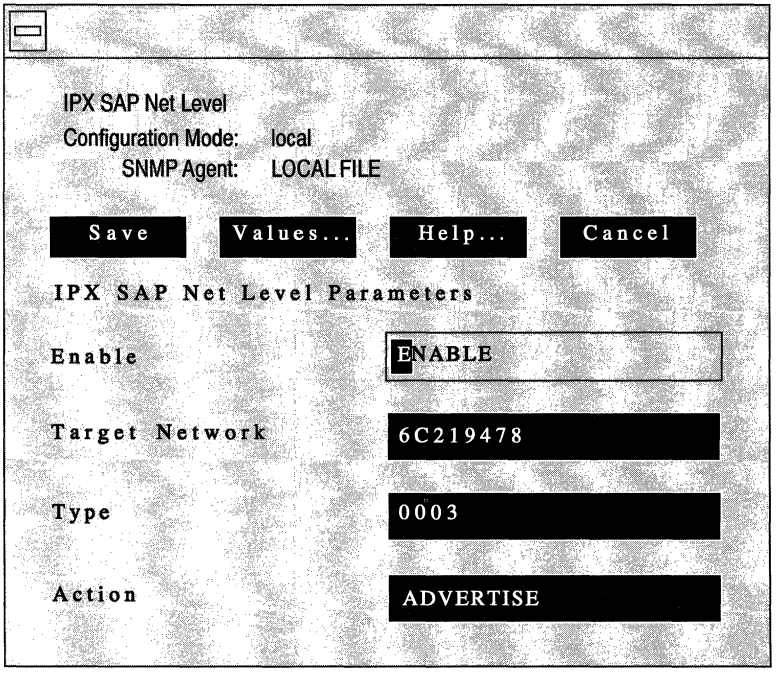


Figure 12-25. IPX SAP Network Level Parameters Window

3. Edit the parameters you wish to change.
4. Click on the Save button to exit the window and save your changes.

The parameters in this window are as follows.

Parameter:	**Enable**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether the SAP network level filter displayed is active on this interface.
Instructions:	Select Enable to enable the SAP network level filter. Select Disable to disable the SAP network level filter.

Parameter:	**Target Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the address of the network to which you wish to configure the filter.
Instructions:	Enter a network address of up to eight hexadecimal characters. You can specify all networks by entering FFFFFFFF.

Parameter:	**Type**
Wellfleet Default:	None
Options:	Any valid Novell server type number in 4-digit hexadecimal format.
Function:	Specifies the type of server to monitor.
Instructions:	Enter the server type number in 4-digit hexadecimal format. Include leading zeros. Refer to Table 12-2 in the previous section for a current list of the well-known server types.

Parameter:	**Action**
Wellfleet Default:	Advertise
Options:	Advertise/Suppress
Function:	Specifies how to process the SAP advertisement matching the Network Number and server Type pattern.
Instructions:	Select Advertise to configure the IPX router to route SAP advertisements matching the Network Number and server type pattern. Select Suppress to configure the IPX router to drop SAP advertisements matching the Network Number and server type pattern.

Deleting a Network Level Sap Filter

To delete a Network Level Sap Filter, first select the filter you wish to delete in the IPX SAP Network Level Window (see Figure 12-23). Then click on the Delete button to display the Delete IPX SAP Network Level Window. Click on the Delete button to delete the filter.

Editing Server Level SAP Filter Parameters

The Wellfleet server level SAP filters function allows you to reduce network traffic by configuring server level SAP filters.

You can add, edit, and delete network level SAP filters for each interface. You perform these functions from the IPX SAP Server Level Window (see Figure 12-26). Begin at the Wellfleet Configuration Manager Window and select the Protocols/IPX/Sap Server Levels option. The IPX SAP Server Level Window appears.

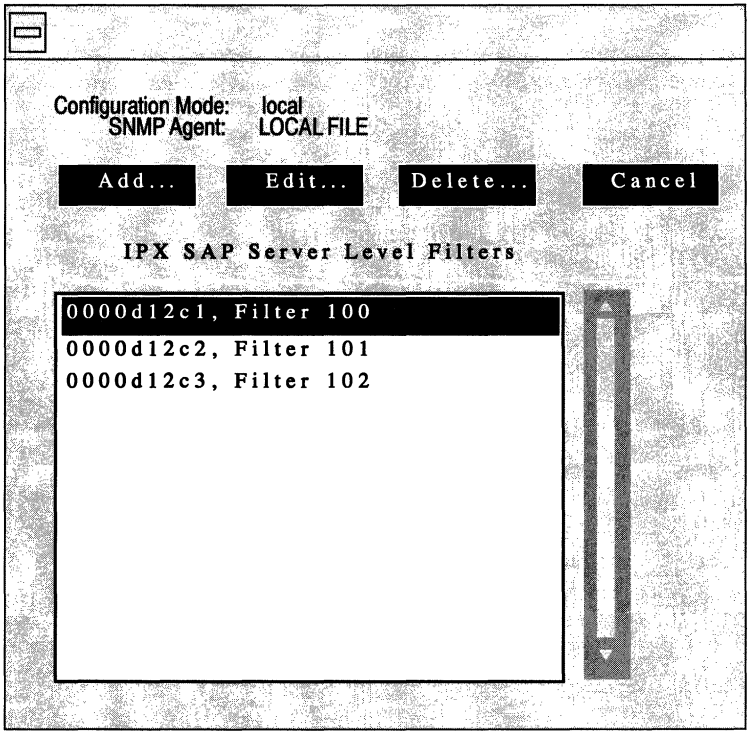


Figure 12-26. IPX SAP Server Level Window

The IPX SAP Server Level Window displays each server level SAP filter you add in hexadecimal notation in the following order:

<Interface No.>, Filter <Filter No.>

The interface and filter numbers are assigned by the system automatically.

Adding a Server Level SAP Filter

To add a server level SAP filter, begin at the IPX SAP Server Level Window (see Figure 12-26) and proceed as follows:

1. Click on the Add button.

   The Add SAP Server Filters Window appears (see Figure 12-27). This window contains the parameters required to add a network level SAP filter. (The parameters with default values are described in the next section.)

2. Edit those parameters you wish to change.
3. Click the Save button to save your changes and exit the window.

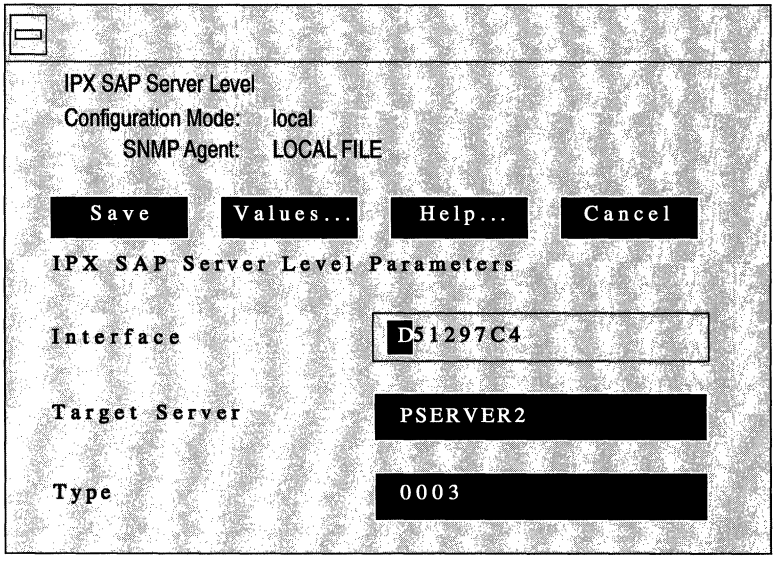


Figure 12-27. Add SAP Server Level Filters Window

The parameters in this window are as follows.

Note: The Site Manager will not allow you to create duplicate server level SAP filters for the same interface.

Parameter :	**Interface**
Wellfleet Default:	None
Options:	Any configured interface address in hexadecimal notation.
Function:	Specifies the address of the interface to which you wish to configure the filter.
Instructions:	Enter an interface address of up to eight hexadecimal characters.

Parameter :	**Target Server**
Wellfleet Default:	None
Options:	Any valid Novell server name from 2 to 47 characters. The name cannot begin with a period, cannot contain a space, and cannot contain the following characters: " * + , / \ | : ; = < > ? [ ]
Function:	Specifies the name of the server for which you wish to configure the filter.
Instructions:	Enter the Novell server name.

Note: The Target Server parameter is case-sensitive. Enter it as it appears in the Novell network.

Parameter: **Type**

Wellfleet Default: None

Options: Any valid Novell server type number in 4-digit hexadecimal format.

Function: Specifies the type of server to monitor.

Instructions: Enter the server type number in 4-digit hexadecimal format. Include leading zeros. Refer to Table 12-3 for a current list of the well-known server types.

Table 12-3. Well-Known Server Types

Server Type	Hexadecimal Identifier
Wild	FFFF (See note below.)
Unknown	0000
Print Server	0003
File Server	0004
Job Server	0005
Archive Server	0009
Remote Bridge Server	0024
Advertising Print Server	0047
Reserved Up To	8000

Note: You **cannot** enter FFFF to specify all server types when configuring server level filters (as you can with network level filters).

Editing a Server Level Sap Filter

You edit a server level SAP filter to change the settings of configurable server level SAP filter parameters, including the default settings.

To edit a server level SAP filter, begin at the IPX SAP Server Level Window (see Figure 12-26) and proceed as follows:

1. Select the filter you wish to edit.
2. Click on the Edit button.

   The IPX SAP Server Level Parameters Window appears (see Figure 12-28).
3. Edit the parameters you wish to change.
4. Click on the Save button to exit the window and save your changes.

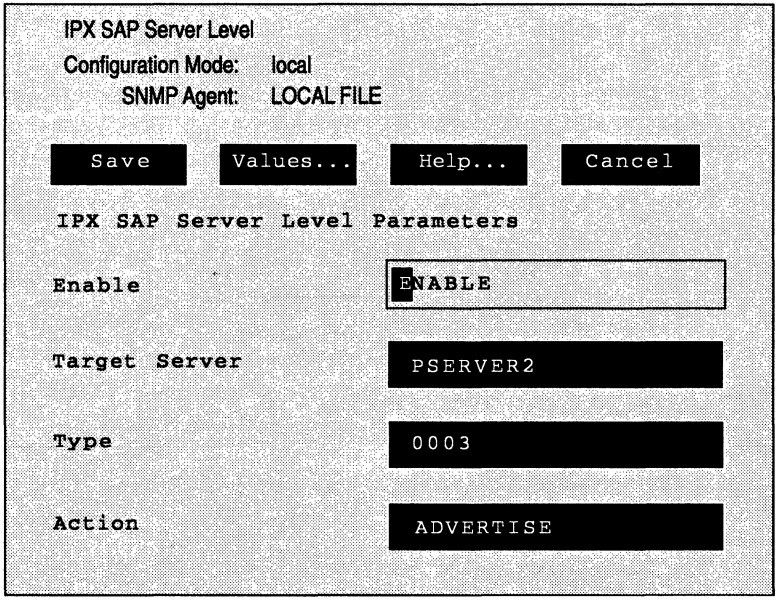


Figure 12-28. IPX SAP Server Level Parameters Window

The parameters in this window are as follows.

Parameter :	**Enable**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether the SAP server level filter displayed is active on this interface.
Instructions:	Select Enable to enable the SAP server level filter. Select Disable to disable the SAP server level filter.

Parameter :	**Target Server**
Wellfleet Default:	None
Options:	Any valid Novell server name from 2 to 47 characters. The name cannot begin with a period, cannot contain a space, and cannot contain the following characters: " * + , / \ | : ; = < > ? [ ]
Function:	Specifies the name of the server for which you wish to configure the filter.
Instructions:	Enter a Novell server name.

Parameter:	Type
Wellfleet Default:	None
Options:	Any valid Novell server type number in 4-digit hexadecimal format.
Function:	Specifies the type of server to monitor.
Instructions:	Enter the server type number in 4-digit hexadecimal format. Include leading zeros. Refer to Table 12-3 in the previous section for a current list of the well-known server types.

Note: You **cannot** enter FFFF to specify all server types when configuring server level filters (as you can with network level filters).

Parameter:	Action
Wellfleet Default:	Advertise
Options:	Advertise/Suppress
Function:	Specifies how to process the SAP advertisement matching the Network Number and Type pattern.
Instructions:	Select Advertise to configure the IPX router to route SAP advertisements matching the Network Number and Type pattern. Select Suppress to configure the IPX router to drop SAP advertisements matching the Network Number and Type pattern.

Deleting a Server Level Sap Filter

To delete a SAP Server Filter, first select the filter you wish to delete in the IPX SAP Server Level Window (see Figure 12-26). Then click on the Delete button to display the Delete IPX SAP Server Level Window. Click on the Delete button to delete the filter.

Deleting IPX from the Wellfleet Router

You can delete IPX from all Wellfleet router interfaces on which it is currently enabled in two steps.

To delete IPX (if it is enabled), begin at the Wellfleet Configuration Manager Window and complete the following steps:

1. Select the Protocols/IPX/Delete IPX option.

   A confirmation window appears.

2. Select OK.

   You are returned to the Wellfleet Configuration Manager window. IPX is no longer configured on the Wellfleet router.

Note: If you deleted IPX, the connectors for those interfaces on which the IPX was the *only* protocol enabled are no longer highlighted in the Wellfleet Configuration Manager Window. Interfaces must be reconfigured for these connectors; see *Configuring Circuits* for instructions.

Chapter 13

Configuring SNMP

List of Figures

List of Tables

Configuring SNMP

About this Chapter

This chapter describes how to configure the Simple Network Management Protocol (SNMP) agent software, which enables the Wellfleet router to respond to Site Manager requests. The first section provides an overview of SNMP. The second section describes how to use the Configuration Manager to edit SNMP parameters.

SNMP Overview

SNMP is a simple request/response protocol that is used to communicate management information between two types of SNMP software entities:

- SNMP Applications (also called SNMP managers)

  Runs in a network management center and issues queries to gather information about the status, configuration, and performance of external network devices (called *network elements* in SNMP terminology). The Site Manager is an example of a network management center, and the Wellfleet router is an example of a network element.

- SNMP Agents

  Runs in network elements (for example, in the Wellfleet router) and responds to network management center queries (for example, from the Site Manager). In addition, if so configured, agent software automatically generates unsolicited reports (called *traps*) back to the network management center when certain significant activity occurs.

For security reasons, the SNMP agent validates each request from an application entity before responding to the request. The validation procedure consists of verifying that the application entity belongs to an SNMP community with access privileges to the agent.

An SNMP community is simply a logical relationship between an SNMP agent and one or more SNMP managers. An SNMP community has a name, and all members of a given community have the same access privileges: either read-only (allows members to view configuration and performance information) or read-write (allows members to view configuration and performance information, as well as change the configuration).

All SNMP message exchanges consist of a community name and the data field, which contains the SNMP operation and its associated operands. You can configure the SNMP agent to receive requests and send responses only from managers that are members of a known community. If the community name in the SNMP message is known to the agent, and the application entity generating the request is known by the agent to be a member of that community, the manager is considered to be authenticated and given the access allowed for members of that community. Thus, the SNMP community allows you to prevent unauthorized managers from viewing or changing the configuration of a Wellfleet router.

SNMP message exchanges are transported by the User Datagram Protocol (UDP); therefore, IP must be enabled in order for SNMP to operate.

Editing SNMP Parameters

Once you have configured a circuit to support IP, you can use the Configuration Manager to edit SNMP parameters. The configuration function you wish to perform, determines the type of parameters you must edit. Table 13-1 lists each configuration function and the section that describes how to perform the function.

Table 13-1. SNMP Parameters and Configuration Functions

To Do the Following:	See this Section:
Change the state of the SNMP agent software. Change how the agent software handles simultaneous requests from more than one network management center. Change if, and when, the agent software generates traps.	*Editing SNMP Global Parameters*
Configure the SNMP communities that have access to the Wellfleet router.	*Editing SNMP Community Parameters*

This section describes how to access and edit SNMP parameters. For each parameter, it provides the following:

- Wellfleet default
- Valid options
- Parameter's function
- Instructions for setting the parameter

To begin, display the Configuration Manager Window (see Figure 13-1); the first window displayed when you enter the Configuration Manager application.

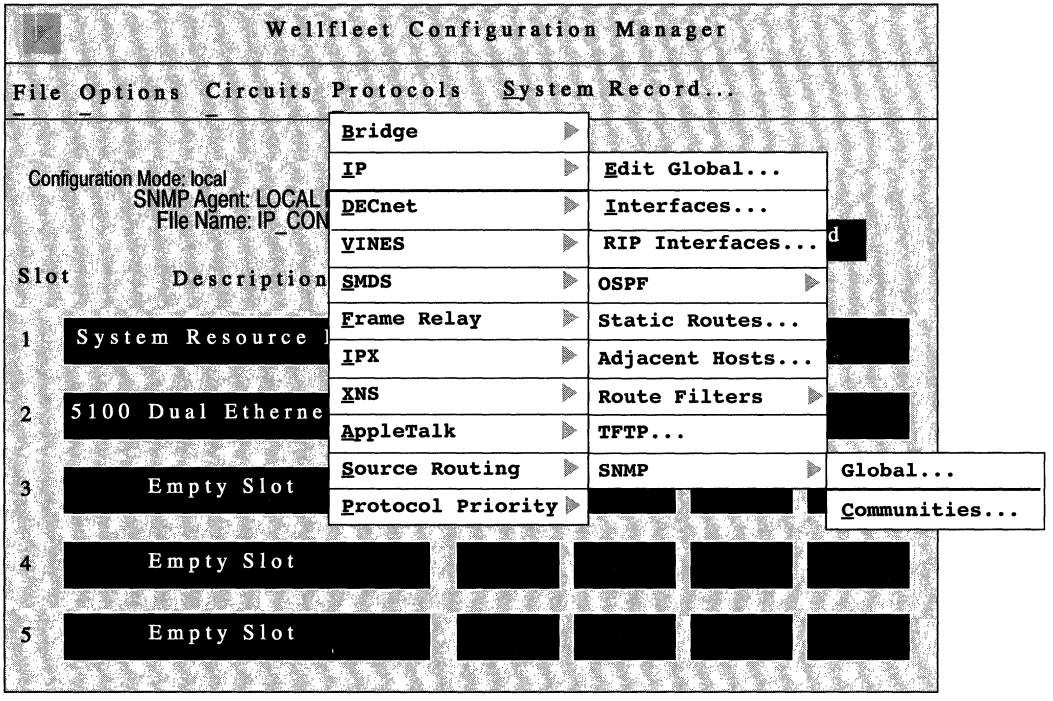


Figure 13-1. Configuration Manager Window

Editing SNMP Global Parameters

You edit SNMP global parameters in the SNMP Global Parameters Window (see Figure 13-2). From the Configuration Manager Window, select the Protocols/IP/SNMP/Global option to display the SNMP Global Parameters Window.

This section provides information you need to edit each parameter in the SNMP Global Parameters Window. Refer to this information to edit the parameters you wish to change. When you are done, click on the Save button to exit the window and save your changes.

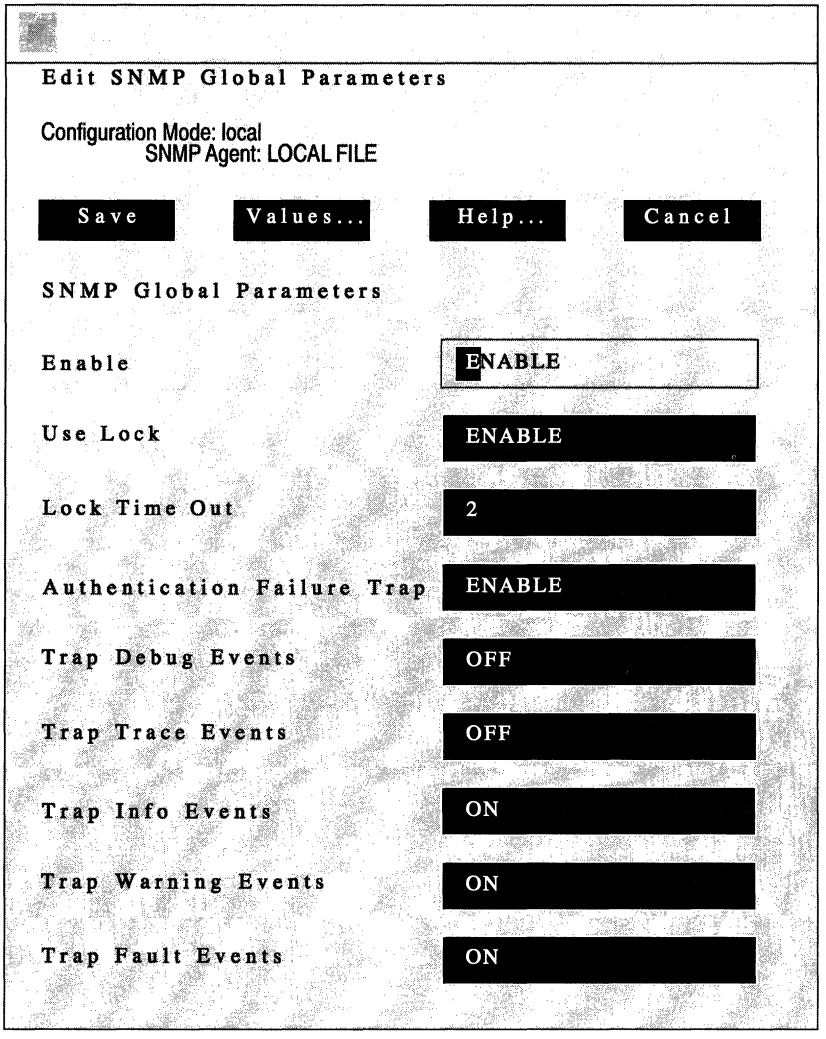


Figure 13-2. SNMP Global Parameters Window

Parameter : **Enable**

Wellfleet Default: Enable

Options: Enable/Disable

Function: Specifies the state of the SNMP agent software on all interfaces that support IP.

Instructions: Select Enable to enable the SNMP agent software.

Select Disable to disable the SNMP agent software.

Warning: When you disable the SNMP agent software in dynamic mode, you prohibit immediately the Site Manager from communicating with the Wellfleet router.

Parameter : **Use Lock**

Wellfleet Default: Enable

Options: Enable/Disable

Function: Specifies whether the agent software responds to multiple network management centers issuing simultaneous SNMP SETs to the Wellfleet router.

Instructions: Select Enable to prohibit the agent software from responding to simultaneous SNMP SETs from multiple network management centers. When you select Enable, the agent software responds to the first SNMP SET it receives and locks out subsequent SETs from other network management centers for the duration of the value you specify at the Lock Time Out parameter. During this lock out time, the agent software will respond to SETs from the network management center which holds the lock; however, it will respond to SETs from other managers with an SNMP genErr GetResponse PDU.

Select Disable to allow the IP router to respond to simultaneous SETs from multiple network management centers.

Parameter :	**Lock Time Out**
Wellfleet Default:	2 minutes
Options:	1 to 60 minutes
Function:	Specifies the maximum number of minutes the IP router allows an idle network management center to hold a lock on the Wellfleet router. During this time, the agent locks out SNMP SETs from other network management centers. The lock timer is reset each time the locking manager issues a SET.
Instructions:	Enter the maximum number of minutes only if you set Use Lock to Enable.

Parameter :	**Authentication Failure Trap**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether the IP router attempts to generate an Authentication Failure Trap when it receives an SNMP message from an SNMP manager falsely claiming to be in a particular community or specifying an unknown community.
Instructions:	Select Enable to enable the IP Router to generate Authentication Failure Traps. If you select Enable, you must configure an SNMP manager to receive the trap (see *Configuring SNMP Communities*). Select Disable to prohibit the IP Router from generating Authentication Failure Traps.

Parameter:	**Trap Debug Events**
Wellfleet Default:	Off
Options:	On/Off
Function:	Specifies whether the IP Router generates Debug Event Traps. Debug Event Traps are used by Wellfleet customer support personnel only.
Instructions:	Select On to enable the IP Router to generate Debug Event Traps. If you select On, you must configure an SNMP manager to receive the traps (see *Configuring SNMP Communities*). Select Off to prohibit the IP Router from generating Debug Event Traps.

Parameter:	**Trap Trace Events**
Wellfleet Default:	Off
Options:	On/Off
Function:	Specifies whether the IP Router generates a Trace Events Trap for every packet it transmits. A Trace Event Trap contains protocol-specific information about the packet.
Instructions:	Select On to enable the IP Router to generate Trace Event Traps. If you select On, you must configure an SNMP manager to receive the traps (see *Configuring SNMP Communities*). Select Off to prohibit the IP Router from generating Trace Event Traps.

Parameter : **Trap Info Events**

Wellfleet Default: On

Options: On/Off

Function: Specifies whether the IP Router generates an Info Event Trap when a routine event occurs (for example, when a service initializes).

Instructions: Select On to enable the IP router to generate Info Event Traps. If you select On, you must configure an SNMP manager to receive the traps (see *Configuring SNMP Communities*).

Select Off to prohibit the IP router from generating Info Event Traps.

Parameter : **Trap Warning Events**

Wellfleet Default: On

Options: On/Off

Function: Specifies whether the IP Router generates Warning Event Traps when a service behaves in an unexpected fashion.

Instructions: Select On to enable the IP router to generate Warning Event Traps. If you select On, you must configure an SNMP manager to receive the traps (see *Configuring SNMP Communities*).

Select Off to prohibit the IP router from generating Warning Event Traps.

Parameter :	**Trap Fault Events**
Wellfleet Default:	On
Options:	On/Off
Function:	Specifies whether the IP Router generates a Fault Event Trap when there is a major disruption in service that could have been caused by a configuration, network, or hardware problem.
Instructions:	Select On to enable the IP Router to generate Fault Event Traps. If you select On, you must configure an SNMP manager to receive the traps (see *Configuring SNMP Communities*). Select Off to prohibit the IP Router from generating Fault Event Traps.

Editing SNMP Community Parameters

The SNMP Community List Window (Figure 13-3) allows you to add, edit, and delete the SNMP communities to which the SNMP agent responds or sends traps.

Note: During initialization, if the SNMP agent detects no valid community with at least one manager, the agent automatically configures a read-write public community with a wild card manager (0.0.0.0) — if the public community already exists, the agent adds the wild card manager to it. Therefore, the Wellfleet router is always SNMP manageable.

For security reasons, Wellfleet recommends that you replace the public community and wild card manager with a unique community configured with a limited list of managers.

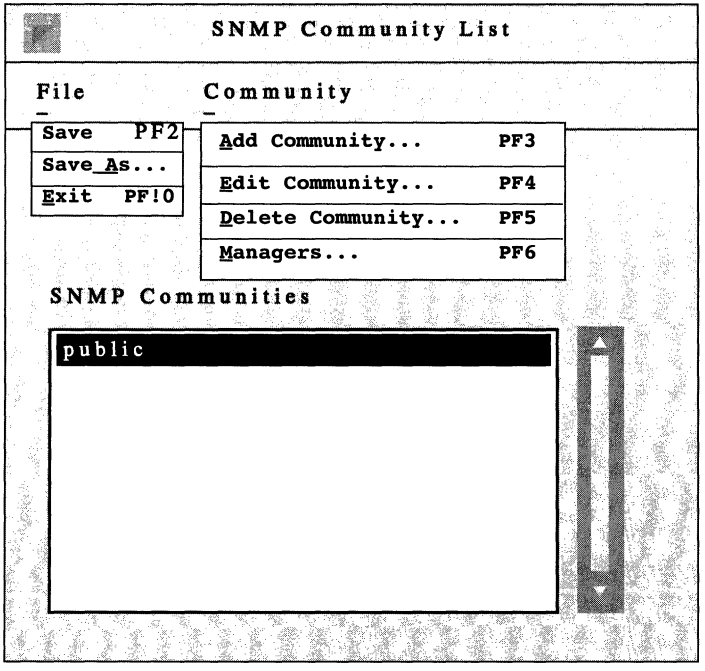


Figure 13-3. SNMP Community List Window

The following sections describe how to add, edit, and delete SNMP communities. To begin, in the Configuration Manager Window, select the Protocols/IP option/SNMP/Communities to the SNMP Community List Window.

Adding an SNMP Community

To add an SNMP community, select the Community/Add Community option in the SNMP Community List Window to display the SNMP Community Window (see Figure 13-4).

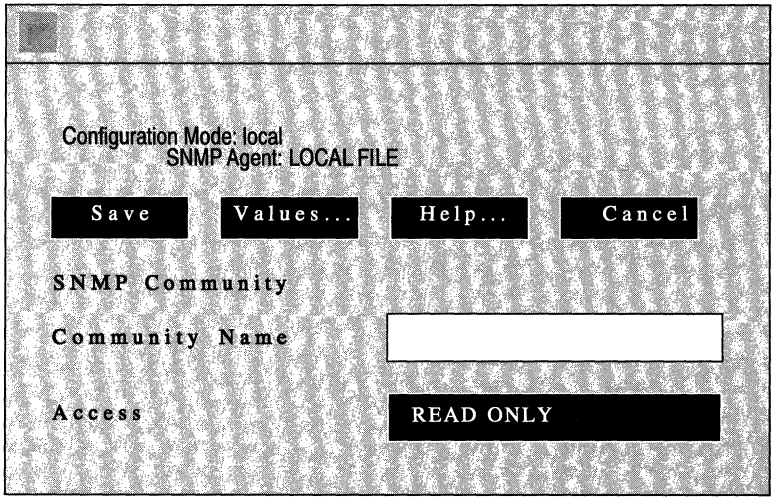


Figure 13-4. SNMP Community Window

Refer to the following parameter descriptions to enter the required information and then click on the Save button to add the SNMP community. After you add the community, you must specify the members of that community; see *Configuring Managers*.

Parameter :	**Community Name**
Wellfleet Default:	None
Options:	Printable ASCII Characters
Function:	Specifies the name of the SNMP community.
Instructions:	Enter the SNMP community name.

Parameter:	Access
Wellfleet Default:	Read Only
Options:	Read Only/Read-Write
Function:	Specifies the access privileges that the Wellfleet router grants to all members of this SNMP community.
Instructions:	Select Read Only to allow all members of this community to view only configuration and performance information about this Wellfleet router. Select Read-Write to allow all members of this community both to view configuration and performance information about this Wellfleet router, as well as to change the Wellfleet router's configuration.

Editing an SNMP Community

To edit an SNMP community, first select the community you wish to edit in the SNMP Community List Window, and then select the Community/Edit Community option to display the SNMP Community Window for that community (see Figure 13-4). The Configuration Manager allows you to change both the name and access privilege for a particular community. See *Adding an SNMP Community* for instructions on how to configure these parameters. If you wish to add, edit, or delete community members from this community, see *Configuring Managers*.

Deleting an SNMP Community

To delete an SNMP community, first select the community you wish to delete in the SNMP Community List Window, and then select the Community/Delete Community option to display the Delete SNMP Community Window (see Figure 13-5). Verify that the proper SNMP community name is displayed and click on the Delete button to delete the SNMP community.

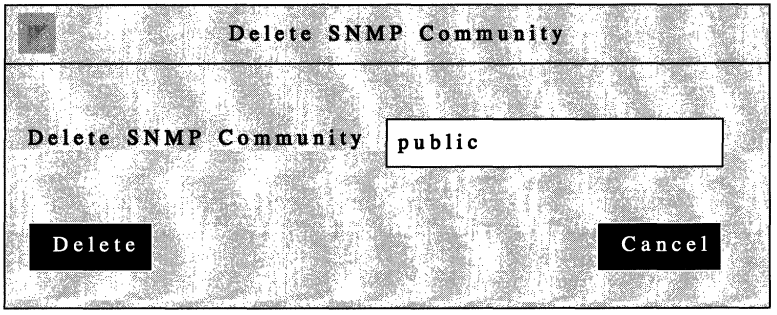


Figure 13-5. Delete SNMP Community

Configuring Managers

The SNMP Manager List Window (Figure 13-6) allows you to add, edit, and delete a particular SNMP community's members (called managers). The following sections describe each procedure. To begin, in the SNMP Community List Window, select the SNMP community for which you wish to configure managers, and then select the Community/Managers option to display the SNMP Manager List Window for that community.

Note: During initialization, if the SNMP agent detects no valid community with at least one manager, the agent automatically configures a read-write public community with a wild card manager (0.0.0.0) — if the public community already exists, the agent adds the wild card manager to it. Therefore, the Wellfleet router is always SNMP manageable.

For security reasons, Wellfleet recommends that you replace the public community and wild card manager with a unique community configured with a limited list of managers.

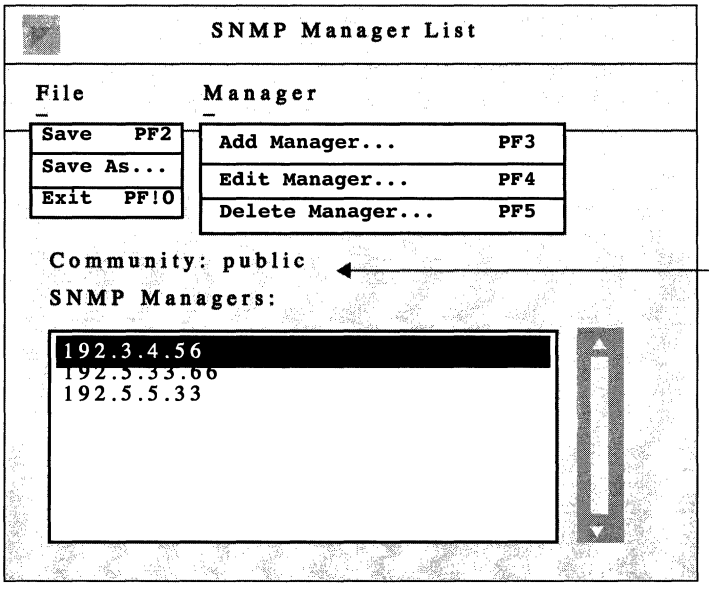


Figure 13-6. SNMP Manager List

Adding a Manager

You add a manager to an SNMP community in the Add SNMP Manager Window (see Figure 13-7). To display this window, select the Community/Add Manager option in the SNMP Manager List Window. Enter the IP address of the SNMP manager and click on the Save button. You must now configure the Manager to receive traps from the Wellfleet router agent software; see *Editing a Manager*.

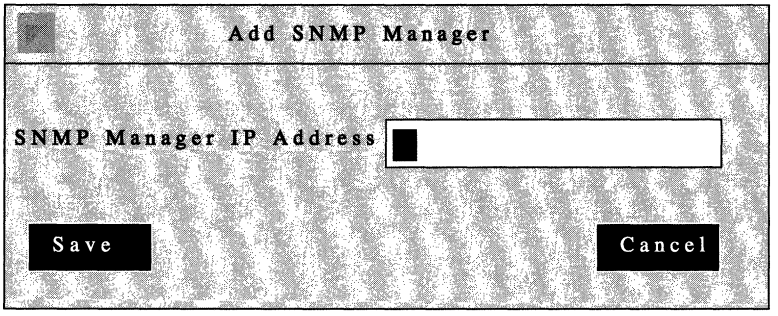


Figure 13-7. Add SNMP Manager Window

Editing a Manager

When you edit a manager, you configure whether and what types of traps the Wellfleet router agent software transmits to that manager. You edit a manager in the SNMP Manager Window (see Figure 13-8). To display this window for a manager, first select the manager in the SNMP Manager List Window and then select Manager/Edit Manager option.

Refer to the following parameter descriptions to enter the required information. When you are done, click on the Save button.

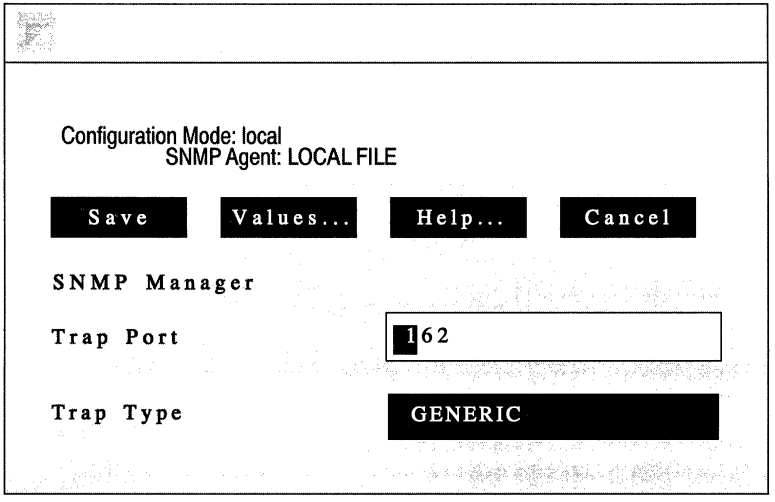


Figure 13-8. SNMP Manager Window

Parameter : Trap Port

Wellfleet Default: 162

Options: 1 to 255

Function: Specifies the number of the port on the managing station to which the agent software transmits traps.

Instructions: The standard port number for trap messages is 162; however, you may enter a different port number. Be sure not to specify a port that is used by another application.

Parameter :	**Trap Type**
Wellfleet Default:	Generic
Options:	None/Generic/Specific/All
Function:	Specifies the type of traps the agent software transmits to this manager.
Instructions:	Select None to prohibit the agent software from transmitting traps to this manager.
	Select Generic to configure the agent software to transmit the well defined SNMP traps (cold start, warm start, and Authentication Failure Traps) to the manager. The well defined cold start and warm start traps are automatically enabled in the SNMP agent software; however, you must enable the Authentication Failure Trap parameter for the agent software to transmit such traps to this manager.
	Select Specific to configure the agent software to transmit all log event traps that you have enabled (for example, the Trap Debug Events, Trap Trace Events, Trap Info Events, Trap Warning Events, and Trap Fault Events parameters) to this manager.
	Select All to transmit cold start and warm start traps, as well as all traps that you have enabled (the Authentication Failure Traps, Trap Debug Events, Trap Trace Events, Trap Info Events, Trap Warning Events, and Trap Fault Events parameters) to this manager.

Deleting a Manager

To delete a manager from an SNMP community, first select the manager you wish to delete in the SNMP Managers List Window for that community, then select the Manager/Delete Manager option to display the Delete SNMP Manager Window. Verify that the proper manager IP address is displayed and then click on the Delete button.

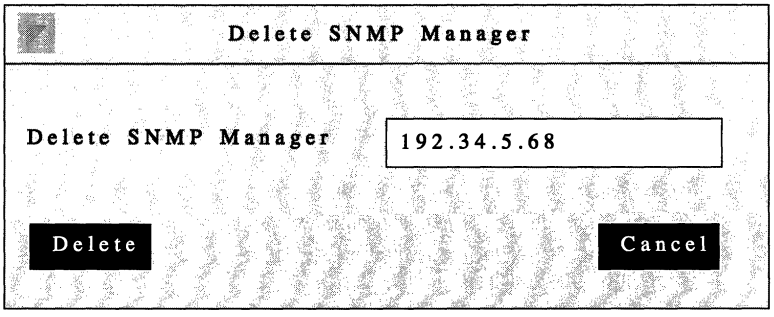


Figure 13-9. Delete SNMP Manager Window

Chapter 14

Configuring VINES

List of Figures

List of Tables

Configuring VINES

About this Chapter

This chapter describes how to configure the VINES router. The first section provides an overview of VINES technology. The second section describes how the Wellfleet VINES router works. The third section lists additional VINES reference material. The fourth section describes implementation guidelines for adding VINES routers to your network. The final sections describe how to use the Configuration Manager to edit VINES parameters and how to delete VINES from the Wellfleet router.

VINES Overview

Banyan VINES (Virtual Networking System) software was developed to provide support for networking personal computers.

Based upon UNIX System 5.3, VINES employs a distributed system environment that allows PC users to communicate and share hardware (printers, disk space, modems etc.) and software (files, applications) resources on a network easily and transparently.

A VINES network consists of servers, clients, and various communications hardware connected via LANs and WANs:

- *Servers* are simply computers that run VINES server software and provide connectivity and services, such as file and print services, to PC users (see Figure 14-1).
- *Clients* are personal computers (PCs) that run VINES client software and use the services provided by servers on the network.

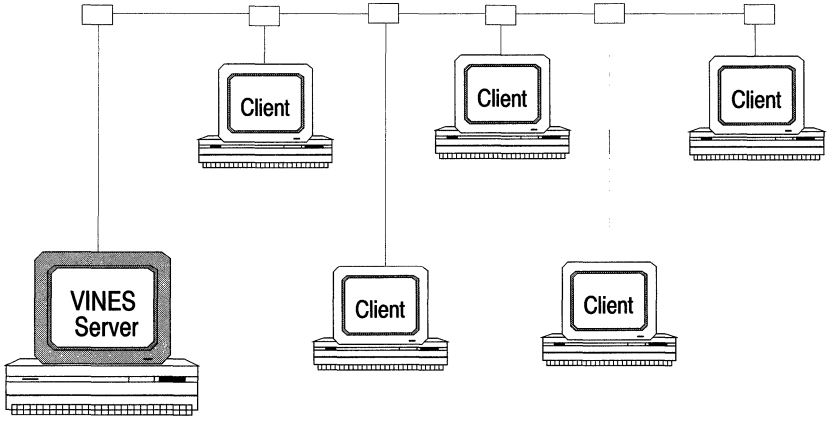


Figure 14-1. VINES Server and Its Clients

To a PC user, VINES presents a complex multivendor network as a single vendor network. Resources from all servers on the network are available to individual PCs; resources appear as though they extend from each individual PC. System administrators control access to these resources. The VINES server hardware can be manufactured by Banyan or one of several other vendors.

Information is routed within a VINES environment as a datagram called a *VINES internet packet*. Each VINES internet packet contains the source and destination address requirements that allow the packet to be routed between nodes on the network. Each VINES internet packet is a discrete unit of data that travels independently on the network layer.

VINES Architecture

The VINES architecture reflects the International Standards Organization for Open System's Interconnection (OSI). The VINES protocol stack is based on seven layers; the lower three layers are dedicated to data delivery and routing while the upper layers are responsible for application specific processes. As shown in Figure 14-2, VINES is designed to support both existing and future OSI model requirements.

Application	VINES Services, VINES Tasker, Unix, and DOS, StreetTalk
Presentation	VINES Matchmaker Data Type Representations
Session	VINES Matchmaker Remote Procedure Calls
Transport	VINES Interprocess Communications Protocol (ICP) VINES Sequenced Packet Protocol (SPP) TCP, UDP
Network	VINES Internet Protocol VINES Internet Control Protocol VINES Address Resolution Protocol VINES Routing Update Protocol X.25, X.3, X.29, IP used by TCP, ICMP, NETBIOS
Data Link	VINES Fragmentation Protocol Drivers for Block Asynchronous, HDLC, Token Ring, Ethernet, other LANS, IEEE 802.x Standards
Physical	Broadband, Baseband, Point-to-Point, Twisted Pair

Figure 14-2. VINES Protocol Stack

At the data link level, VINES currently supports several IEEE standards including Ethernet, Token Ring, and 802.x. In addition, VINES provides its own proprietary data link protocol, called VINES Fragmentation Protocol, which breaks up and reassembles packets that are too large to travel over certain media.

At the networking level, VINES supports both industry standard protocols such as TCP/IP, X.25 and AppleTalk, as well as its own set of networking protocols. These include:

- VINES Internet Protocol (IP)
- VINES Routing Update Protocol (RTP)
- VINES Address Resolution Protocol (ARP)
- VINES Internet Control Protocol (ICP)

Upper layer protocols include VINES print and file service applications and the VINES naming protocol called StreetTalk.

StreetTalk is a distributed directory service that contains the names and attributes of all critical network resources. Each resource on a VINES network is assigned a StreetTalk name that is globally unique. StreetTalk names are in the format:

item @ group @ organization

where:

— item identifies a user or resource on the network.

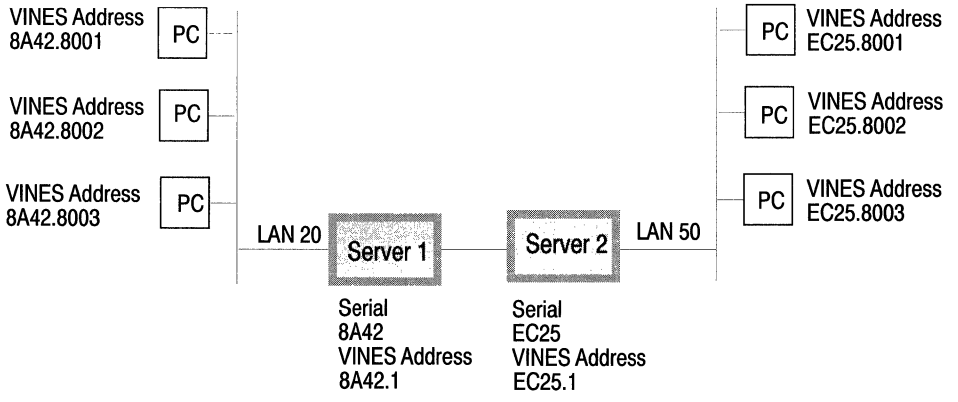
— group identifies the group to which the item belongs.

— organization identifies the organization to which the group belongs.

VINES is designed to adapt to changes in the network topology; because each resource is referenced by name, it can be moved or replaced and still located by PCs. System administrators control which resources can be accessed by end users on the network.

VINES Network Addressing

Each grouping of nodes on a VINES network consists of a service node and the client nodes, to which the service node provides address resolution and routing services (see Figure 14-3). Note that this is a logical grouping; client nodes may or may not map directly to the same physical media.



LAN = LAN Address
Serial = Serial Number
VINES Address = Network Number.Subnetwork Number

Figure 14-3. VINES Network

When a client node becomes active on the network, it broadcasts a query request for all servers. All reachable servers respond. The client node chooses the first server that responds and requests a VINES internet address from that service. The service node assigns a unique, 48-bit VINES internet address to the client node.

The VINES internet address is independent of any data link layer address assigned to a node on a physical medium. The 48-bit VINES internet addresses consist of two fields (see Figure 14-4):

- The 32-bit network number field. The network number is the serial number of the server node and identifies the logical grouping of nodes on a VINES network.
- The 16-bit subnetwork number field. The subnetwork number identifies the node within the server node's logical grouping.

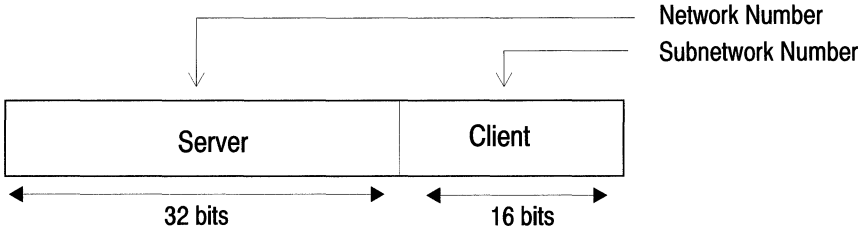


Figure 14-4. VINES Internet Address Format

The internet address for each service node in a VINES network is its network number, concatenated with the subnetwork number of 1. The service node assigns unique internet addresses to all other client nodes in its subnetwork by concatenating its network number with a unique subnetwork number for each node. Subnetwork numbers are assigned as follows:

Subnetwork Number(s):	Used for Node Type:
1	For Server only
2 - 0x7fff	Unused
0x8000 - 0xfffe	Clients only
ffff	Broadcast

How the Wellfleet VINES Router Works

On a VINES network, the Wellfleet VINES router maintains the network topology and uses both IEEE standard and VINES proprietary protocols to route packets through the network. The Wellfleet VINES router supplies client nodes with addresses only if there are no other servers on the network.

The following sections describe the VINES data link and routing protocols used by the Wellfleet VINES router.

VINES Data Link Protocols

The VINES data link layer protocols support the exchange of data frames between neighboring server and client nodes on the network. Frames can be broadcast. The maximum frame size is 1500 bytes.

In addition to supporting most of the IEEE standards, the VINES Fragmentation Protocol breaks up and reassembles packets that are too large to travel over certain media into smaller sized frames. For example, if a node on an Ethernet LAN attempts to send a packet to a node on an IBM PC LAN over a Synchronous network, the data link entity fragments the packet into smaller size frames that can be transported on an IBM LAN.

The fragmentation protocol information is stored in its own two-byte header and follows the data link header in a VINES frame (see Figure 14-5). The first byte contains a sequence number; the second byte contains a control field. The value of the sequence number field is of modulo 256 and is determined by the node that originated the frame. The receiving node uses the number to determine the correct order to reassemble the data packet as fragments are received. If the fragments are received out of order, the intermediate node that reassembles the fragments will discard all fragments. The value of the control field indicates whether the frame begins or ends a VINES IP packet. Only the first fragment includes the VINES IP header.

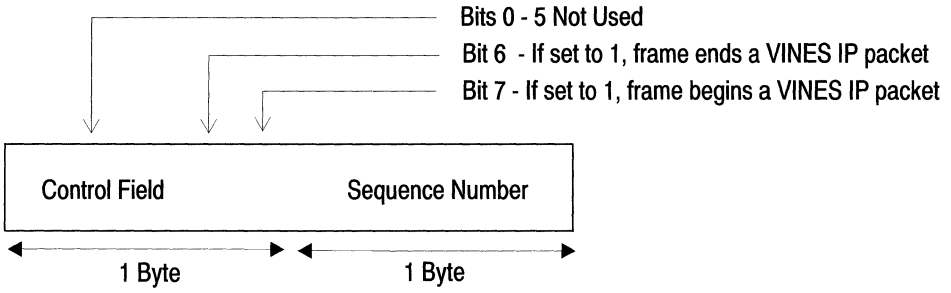


Figure 14-5. VINES Fragmentation Protocol Header

VINES Routing Protocols

The VINES network layer protocols are responsible for routing VINES data packets to destination nodes using the fastest route available. They also distribute the current network topology throughout the network. The VINES network layer supports connectionless (datagram) services only.

The maximum packet length on a VINES network is 1500 bytes, including the VINES internet Protocol header.

The following sections describe Wellfleet implementation of the VINES networking protocols. These include:

- VINES Internet Protocol (IP)
- VINES Routing Update Protocol (RTP)
- VINES Address Resolution Protocol (ARP)
- VINES Internet Control Protocol (ICP)

VINES Internet Protocol

VINES IP is responsible for routing packets from the source node to the destination node whose internet address is specified in the packet header. All internet packets begin with a VINES IP header that identifies the source and destination node addresses, an identifier for the next protocol in the packet, a transport control byte, the length of the entire packet and a software checksum if needed. A header for another network layer protocol or transport layer protocol follows the VINES IP header (see Figure 14-6).

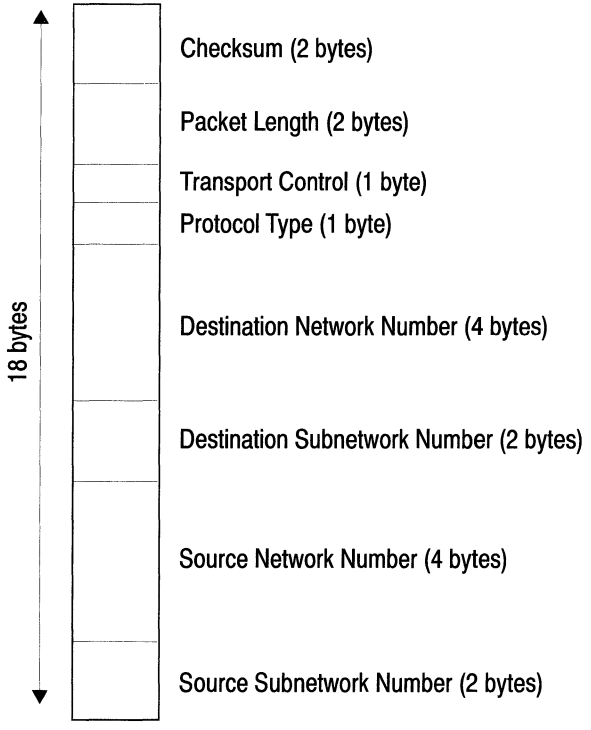


Figure 14-6. VINES Internet Protocol Header

When the Wellfleet router receives a packet, the VINES IP entity on the router handles the packet differently, depending on how the packet is addressed:

- Packets destined for the router

  When the router receives a packet addressed to itself, it first reassembles the packet, if it is fragmented. Next, it makes certain that the packet is not corrupt by checking the checksum, if there is one. Finally, it passes the packet up to the next level for processing.

- Broadcast packets

  When the router receives a broadcast packet, it checks the packet's hop count to make certain that it isn't zero. In most cases if the hop count is zero, the packet is discarded (unless it is a StreetTalk or Time Sync Service packet; see below). If the packet is accepted by the node, it decrements the hop count by one before retransmitting the packet on all interfaces except for that on which it was received.

  There are two exceptions to how the router handles broadcast packets. If the broadcast packet is a *StreetTalk broadcast packet*, which propagates StreetTalk information, or a *Time Sync Service broadcast packet*, which propagates time information, the router ignores the hop count. First, it checks to see if the packet was received on the interface that provides the best path back to the originating node. If so, it retransmitts the packet on all other interfaces (without modifying the hop count field). If not, the router determines that the packet has looped back and the packet is discarded.

  The router recognizes a StreetTalk broadcast packet or a Time Sync Service broadcast packet by examining the destination port field of the packet's ICP header. The destination port field is set to 0x0000F for all StreetTalk packets and to 0x0007 for all Time Sync Service packets.

- Nonbroadcast packets

  When the router receives a nonbroadcast packet with a different destination address, the router knows that it must forward the packet. First, it reassembles the packet if necessary. Next, it refers to its next hop routing table to determine the next hop. Finally, it forwards the packet toward this hop.

VINES Routing Update Protocol

The VINES Routing Update Protocol (RTP) is responsible for maintaining a local routing table that VINES IP can refer to when selecting paths. RTP is also responsible for distributing this information about the network topology among the servers and clients in the network.

RTP packets have a 4 byte header that immediately follows the VINES IP header (see Figure 14-7).

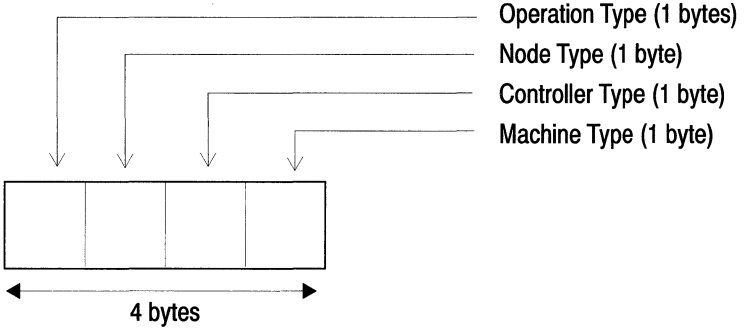


Figure 14-7. VINES Routing Update Protocol Header

The four fields included in the header are as follows:

- *Operation type* specifies whether the packet is a routing request packet, routing update packet, routing redirect or routing response packet.
- *Node type* specifies whether a service node or a client node originated the packet.
- *Controller type* specifies whether a single buffer or multibuffer LAN card originated the packet.
- *Machine type* specifies the type of processor that originated the packet.

RTP distinguishes between service nodes and client nodes on the network. Service nodes route packets addressed to other nodes. Service nodes are usually servers. Client nodes do not perform any routing services. Both service nodes and client nodes maintain two routing tables: a *table of all known networks* and a *table of neighbors* (see Table 14-1).

Table 14-1. VINES Routing Tables

Each Entry in this Table:	Contains this Information:
Table of Networks	Network number, routing metric to reach network, next hop used to reach the network
Table of Neighbors	Network number, subnetwork number, medium over which the neighbor can be reached, LAN address of neighbor, routing metric to reach neighbor.

For service nodes, the table of all known networks contains an entry for all known networks, except for the server's own. Client nodes, on the other hand, keep track of only the networks with which they are currently communicating with - thus reducing table space. For service nodes, the table of neighbors contains an entry for each neighboring node. For client nodes, the table of neighbors contains an entry for each neighbor with which they are communicating.

The RTP entities exchange these four types of packets:

- Routing update packets

  Every node on a VINES network periodically broadcasts routing update packets. Client nodes on LAN and high-speed media send out routing update packets every 90 seconds. These packets inform neighbors of the node's existence and the node's type. Routes remain in a neighbor's routing table for 6 minutes. If the route is not heard from after 6 minutes, it will be marked as unreachable and removed from the routing table.

  Service nodes send out routing packets that inform neighbors of its type and existence, and also include a list of all networks known to the service node and the cost of reaching these networks from the service node. However, on server to server connectivity, and over WAN connections (TCP/IP, X.25, HDLC, Block Asynchronous), three full routing update packets are sent out when the node first comes up on the network. Afterwards, routing update packets are only generated when routing changes are made to the network. All routes permanently remain in a node's routing table for these types of connections.

- Routing request packets

  Routing request packets are generated by a client node when it needs information about the network topology.

- Routing response packets

  In response to receiving a routing request packet, service nodes generate routing response packets that describe the network topology.

- Routing redirect packets

  Routing redirect packets are generated by service nodes when a service node determines that it should not be used for forwarding packets between two nodes because a better path exists. The service node then sends a routing redirect packet to the last hop that forwarded the packet, informing it of the existence of a better route. The service node also sends the original packet toward the destination.

VINES Address Resolution Protocol

The VINES Address Resolution Protocol (ARP) allows service nodes to provide address resolution services to client nodes that have not yet been assigned VINES internet addresses. VINES uses the services of VINES IP to deliver address resolution packets between nodes.

A VINES ARP packet is prefixed with an eight-byte header and follows the VINES IP header (see Figure 14-8).

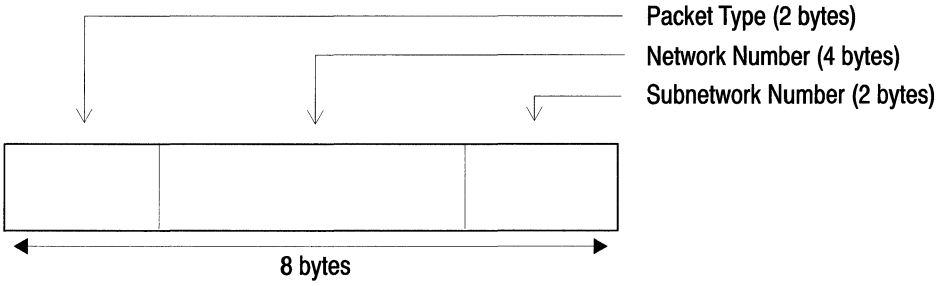


Figure 14-8. VINES Address Resolution Protocol Header

The protocol defines two types of entities:

- Address resolution services

  An address resolution service is implemented within a node that can route VINES packets and has a static, unique, 32-bit network number. Service nodes usually implement address resolution services.

- Address resolution clients

  An address resolution client is implemented in a node that has not been assigned a VINES address. Client nodes usually implement an address resolution client.

VINES Internet Control Protocol

VINES Internet Control Protocol (ICP) provides support to certain transport layer protocol entities by providing notification of some network errors and some topological conditions. A VINES ICP packet is prefixed with a four-byte header and follows the VINES IP header.

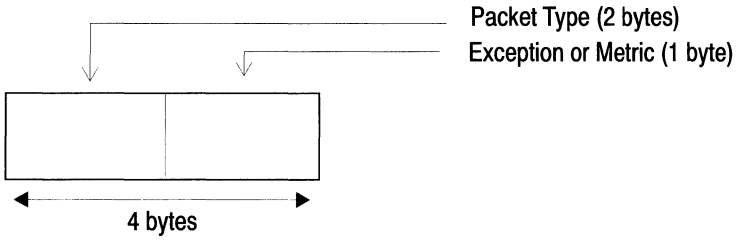


Figure 14-9. VINES Internet Control Protocol Header

Two types of ICP packets are generated by the ICP entity:

- Exception notification packets

  Exception notification packets specify that network layer exceptions occurred during the routing of transport layer messages. The ICP entity generates these packets when:

  — VINES IP can't properly route or receive a VINES IP packet. For example, when a service node receives a packet containing an unknown destination address.

  — The packet has the error subfield enabled in the VINES IP header's transport control field.

- Metric notification packets.

  Metric notification packets contain metric information about the final transmission medium used to reach a client node. The ICP entity generates these packets when:

  — The entity routes a packet with the metric subfield enabled in the transport control field of the VINES IP header.

  — The destination address in the VINES IP header specifies a node that is a neighbor of the service node.

VINES Bibliography

The following documentation provides technical detail on VINES protocol implementation.

VINES Architecture Definition. Banyan Systems Incorporated. August, 1988. 092015-001.

VINES Protocol Definition. Banyan Systems Incorporated. February, 1990. 092093-000.

VINES Implementation Notes

This section provides you with some basic guidelines on adding Wellfleet VINES routers to your network. It also addresses some of the special configuration features that may match your network requirements.

Specifying a VINES Interface Type

When you enable VINES on a circuit, the Interface Type parameter is set to Ethernet by default. If the VINES circuit is of another type (Token Ring or FDDI, for example), make certain to edit the Interface Type parameter so that it matches the type of circuit you are configuring.

See the section entitled *Editing VINES Interface Parameters* for instructions.

Assigning a Network ID to Your Router

When you enable VINES on the router, Wellfleet recommends accepting the default Network ID that the router assigns to itself. However, if you chose to specify a different Network ID make certain the number you assign is unique within the VINES network.

See the section entitled *Editing VINES Global Parameters* for instructions.

Configuring the VINES Router on a Serverless Network Segment

If you enable VINES on a circuit that contains no VINES servers, then you must enable VINES ARP on the circuit so that the router can provide address resolution services to client nodes on this circuit. You enable ARP using the ARP Enable parameter.

See the section entitled *Editing VINES Interface Parameters* for instructions.

In addition, if your VINES network topology is such that there are two or more hops between client nodes and the server that services the circuit, you must do the following:

- Set the circuit's ARP Enable parameter to enable so that the VINES router can provide address resolution services to any client nodes.
- Set the Remote Client Privileges parameter to Enable on those circuits that connect the routers to the server. Doing so allows the client nodes to communicate with the server, even though they are separated by more than one hop.

Note: When you enable Remote Client Privileges, Serverless Networks for WANs is automatically enabled. Banyan does not recommend using Serverless Networks on a WAN because the high cost increases delays and may terminate sessions. Wellfleet, however, does support this configuration.

Figure 14-10 shows a sample VINES network in which the VINES server is separated from the client nodes by two Wellfleet routers. In order for the server and client nodes to communicate, we configured the routers in the following way:

- Enabled VINES on circuits E1, E2, E3, E4.
- Set the ARP Enable parameter to Enable on circuits E3 and E4 so that Router B can provide address resolution services to the client nodes on these circuits.
- Set the Remote Client Privileges parameter to Enable on circuits E1 and E2, so that the client nodes on circuits E3 and E4 can reach the server via Routers A and B.

As a result, the server and client nodes on this network can communicate, even though they are separated by more than one hop.

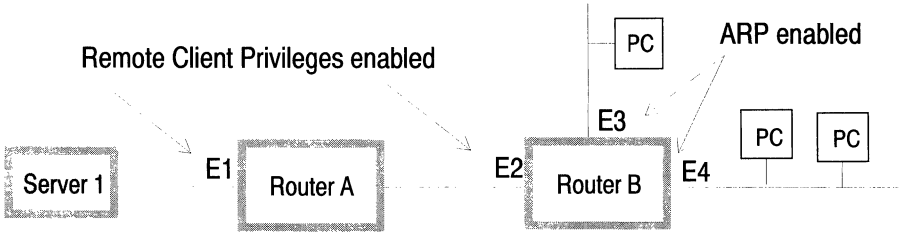


Figure 14-10. VINES Routers Configured on a Serverless Network Segment

Configuring the VINES Router to Source Route Over Token Ring Networks

The Wellfleet VINES router can now route over token ring networks that contain one or more source routing bridges.

In a source routing network, every end station that sends out a frame supplies the frame with the necessary route descriptors so that it can be source routed across the network. Thus, in order for VINES routers to route packets across a source routing network, *they must act like end stations*; supplying route descriptors for each packet before they send it out onto the network.

With end node support enabled, whenever a Wellfleet VINES router receives a packet and determines that the packet's next hop is located across a source routing network, the router does the following:

- Adds the necessary RIF information to the packet's MAC header.
- Sends the packet out onto the network where it is source routed toward the next hop.

Upon receiving the packet from the token ring network, the peer router strips off the RIF field and continues to route the packet toward the destination network address (see Figure 14-11).

You configure source route end node support on a per-circuit basis by setting the End Station Enable parameter to Enable. See the section entitled *Editing VINES Interface Parameters* for instructions on enabling this parameter.

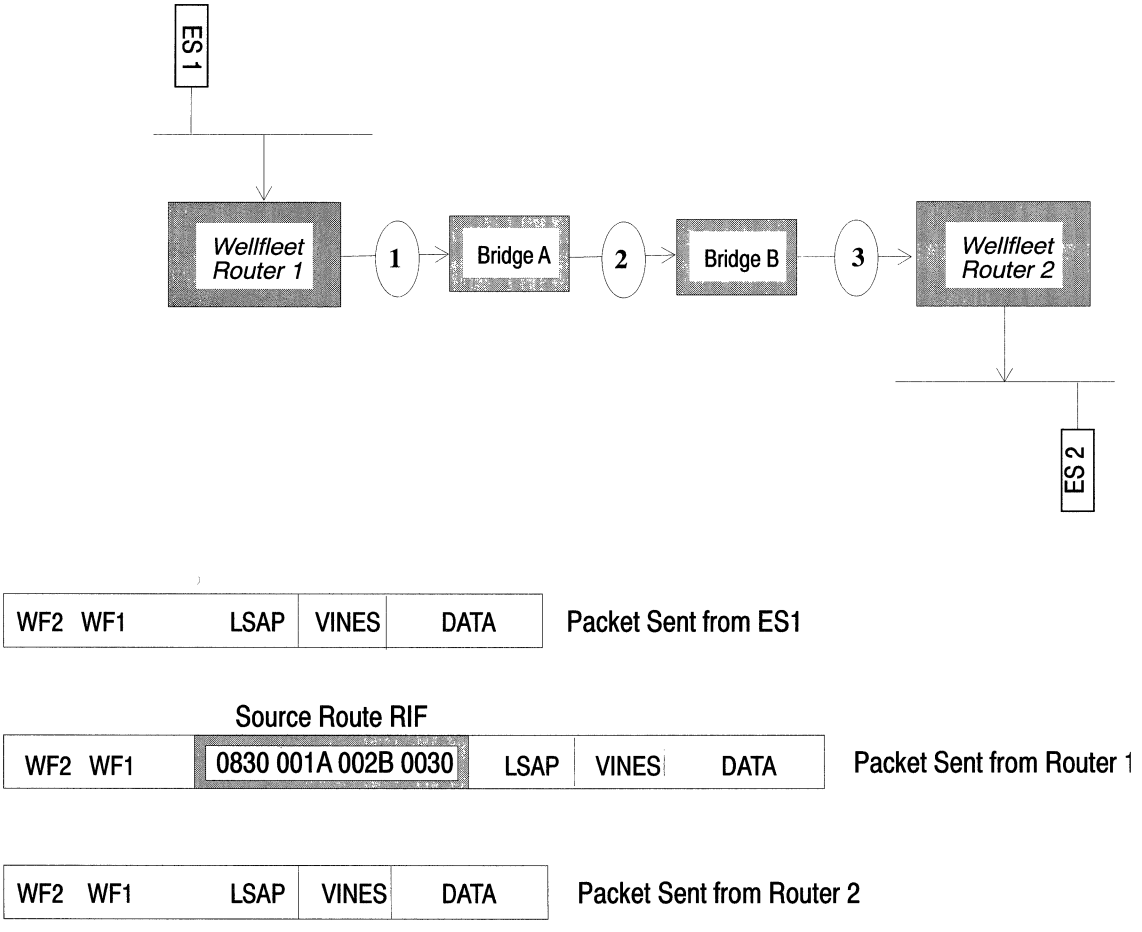


Figure 14-11. VINES Routers Source Routing Across a Token Ring Network

Editing VINES Parameters

Once you have configured a circuit to support VINES, you can use the Configuration Manager to edit VINES parameters. The configuration function you wish to perform determines the type of parameters you must edit. Table 14-2 lists each configuration function and the section that describes how to perform the function.

Table 14-2. VINES Parameters and Configuration Functions

To Do the Following:	See this Section:
Change the state of the VINES router software.	*Editing VINES Global Parameters*
Reconfigure VINES on a particular interface.	*Editing VINES Interface Parameters*
Configure VINES filters.	The chapter *Configuring Filters*

For each VINES parameter, this section provides the following:

- Wellfleet default
- Valid setting options
- Parameter function
- Instructions for setting the parameter

You begin from the Wellfleet Configuration Manager window (see Figure 14-12).

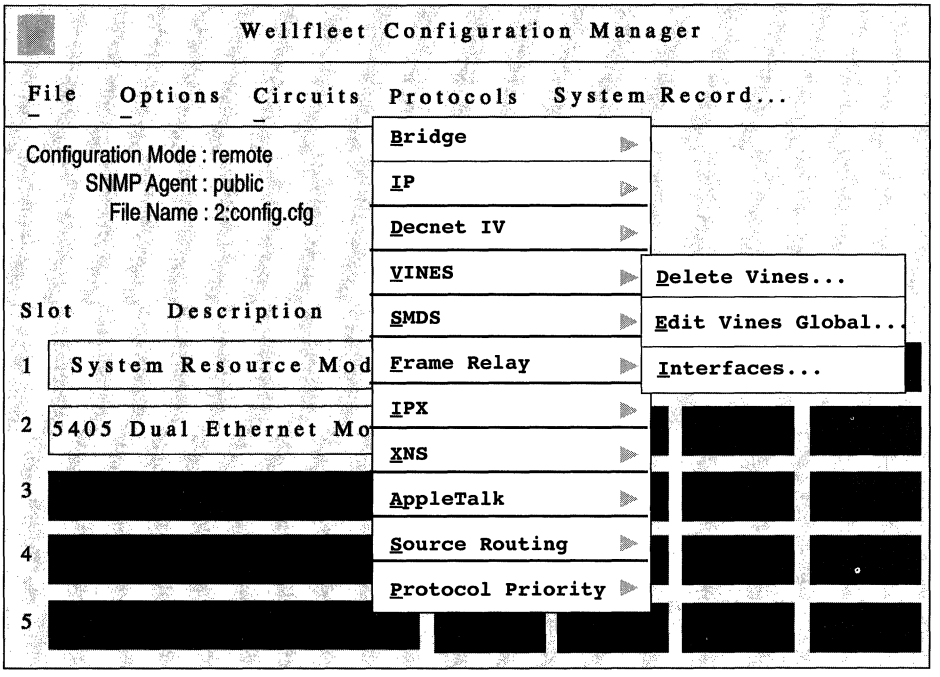


Figure 14-12. Wellfleet Configuration Manager Window

Editing VINES Global Parameters

To edit VINES global parameters, begin at the Wellfleet Configuration Manager window and proceed as follows:

1. Select the Protocols/Vines/Edit Vines Global option.

   The VINES Global Parameters window appears (see Figure 14-13).

2. Edit those parameters you wish to change.

   This section provides the information you need to edit each parameter.

3. Click the Save button to exit the window and save your changes when you are finished.

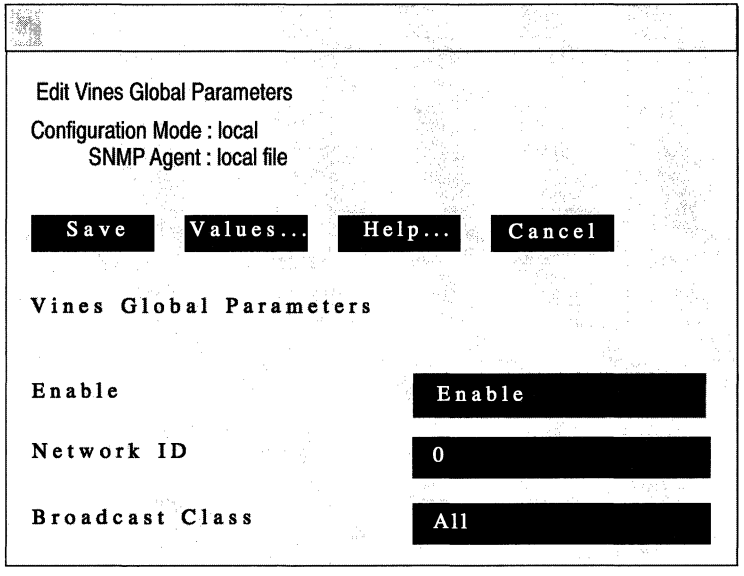


Figure 14-13. VINES Global Parameters Window

Parameter :	**Enable**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Enables or Disables the VINES router on the entire Wellfleet router.
Instructions:	Set to Disable if you want to disable VINES.

Parameter :	**Network ID**
Wellfleet Default:	Variable
Options:	1-2097151
Function:	Specifies the router's Network ID (network number). The Network ID is the 32 bit high order portion of the node's internet address.
Instructions:	Wellfleet strongly recommends accepting the default value that the router assigns. However, if you assign a different Network ID, note the following: ❒ All internet addresses assigned to client nodes in this router's network will begin with the Network ID you specify here. ❒ The Wellfleet VINES router will modify the number you enter here so that the first 11 bits reflects the range of Wellfleet assigned numbers. (For example, if you enter 1 as the Network ID, the router will precede this number with 0x304.)

Parameter : **BroadCast Class**

Wellfleet Default: All

Options: All, No Charge, Low Cost, LANS, Server All, Server No Charge, Server Low Cost, Server LANS

Function: Specifies which nodes residing on the router's interfaces should receive broadcast packets generated by this router. This parameter allows you to control the number of extraneous broadcast packets that nodes receive.

For example, the default class All specifies that all nodes residing on the router's interfaces should receive broadcast packets. In contrast, the class Low specifies that only those nodes on interfaces to which a low cost is associated should receive broadcast packets.

Instructions: Wellfleet recommends accepting the default All. Table 14-3 describes the affect of each broadcast class setting in detail.

Table 14-3. VINES Broadcast Class Description

If Broadcast Class Is:	These Nodes Receive Broadcast Packets:
All	All reachable nodes on any interface
No Charge	All reachable nodes except those on media that impose a packet charge
Low Cost	All reachable nodes residing on low cost media (4800 bits per second serial lines, or faster)
LANS	All reachable nodes on a high-speed media (LANS)
Server All	All reachable service nodes, regardless of media cost
Server No Charge	All reachable service nodes except those residing on media that impose a packet charge
Server Low Cost	All reachable service nodes residing on low cost media (4800 bits per second serial lines, or faster)
Server LANS	All reachable service nodes on a high-speed media (LANS)

Editing VINES Interface Parameters

To edit a VINES interface, begin at the Wellfleet Configuration Manager window then proceed as follows:

1. Select the Protocols/Vines/Interfaces option to display the VINES Interfaces window (see Figure 14-14).

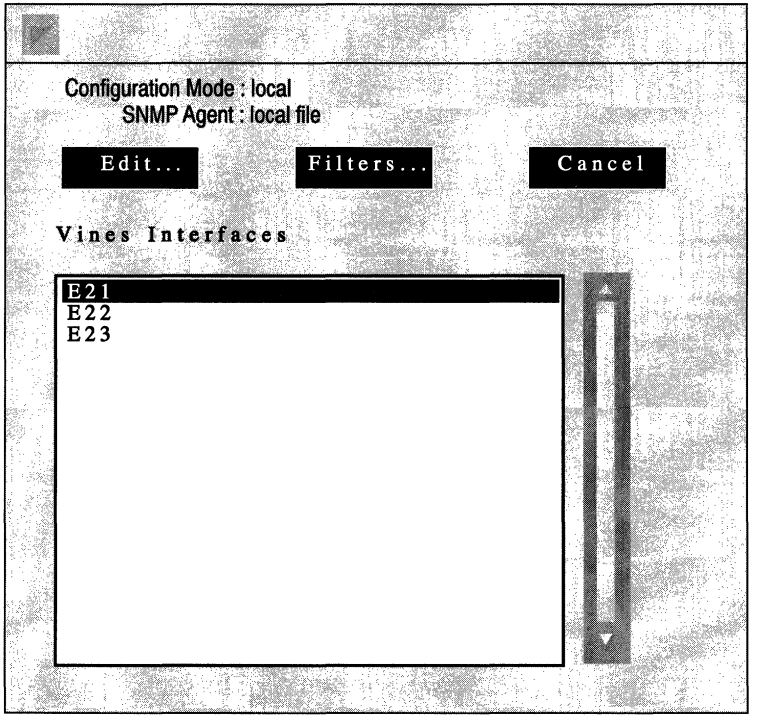


Figure 14-14. VINES Interfaces Window

2. Select the interface you wish to edit.
3. Click on the Edit button to display the VINES Interface Parameters window for that interface (see Figure 14-15).

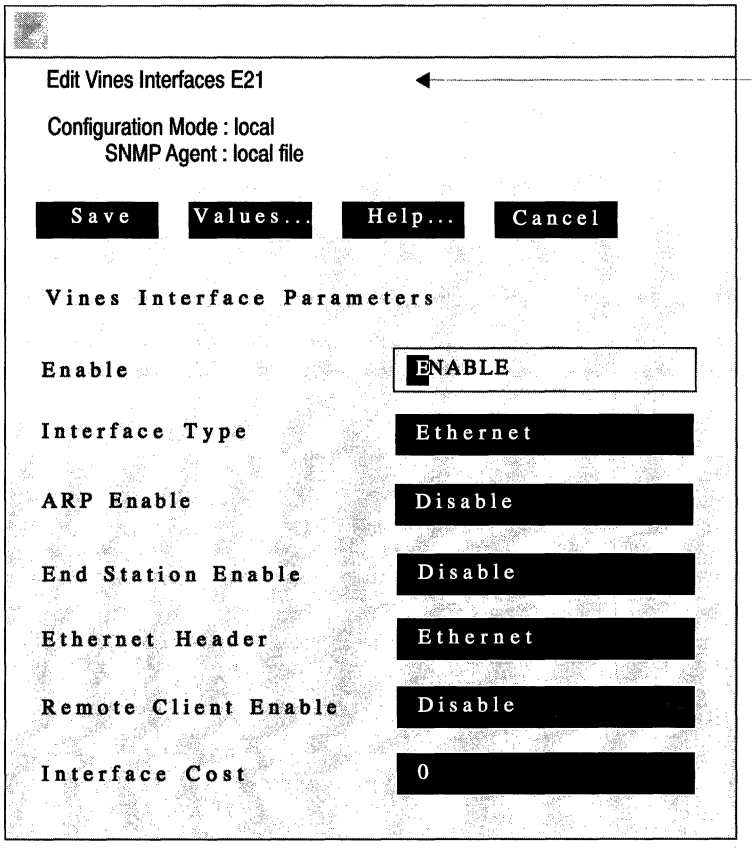


Figure 14-15. VINES Interface Parameters Window

4. Edit those parameters you wish to change.

   This section provides the information you need to edit each parameter.

5. Click the Save button to exit the window and save your changes when you are finished.

Note: When you reconfigure an interface in dynamic mode, VINES restarts on that interface.

Parameter : Enable

Wellfleet Default: Enable

Options: Enable/Disable

Function: Enables or disables VINES over this interface.

Instructions: Disable only if you want VINES disabled over this interface.

Parameter : Interface Type

Wellfleet Default: Ethernet

Options: See Table 14-4.

Function: Defines the interface type and speed.

Instructions: Select the appropriate type that describes this interface as defined in Table 14-4 below.

Table 14-4. VINES Interface Types

Interface Type	Speed Options
Ethernet	---
Token Ring	4M, 16M
HDLC	12K, 48K, 96K, 56K
Async	12K, 48K, 96K, 56K
X.25	12K, 48K, 96K, 56K
T1	45, 128, 192, 256, 320, 384, 448, 512, 576, 640, 704, 896, 1088, 1344
IP Tunnel	---
FDDI	---

Parameter:	**ARP Enable**
Wellfleet Default:	Disable
Options:	Enable/Disable
Function:	Specifies if this interface is enabled to support VINES ARP. With ARP enabled, the router can provide address resolution services to client nodes on this interface that have not yet been assigned addresses.
Instructions:	Enable ARP only if there are no server nodes on this circuit.

Parameter:	**End Station Enable**
Wellfleet Default:	Disable
Options:	Enable/Disable
Function:	Specifies if this interface is enabled for source routing end station support.
Instructions:	Enable if this interface is 1) of type token ring and 2) if source routing is enabled on the VINES servers and clients in this ring.

Parameter:	**Ethernet Header**
Wellfleet Default:	Ethernet
Options:	Ethernet, SNAP
Function:	Specifies the type of encapsulation that this interface supports at the data link level. If this circuit is not an Ethernet circuit, this parameter is ignored.
Instructions:	Accept the default Ethernet. (Future Banyan VINES releases will support SNAP.)

Parameter :	**Remote Client Enable**
Wellfleet Default:	Disable
Options:	Enable/Disable
Function:	When enabled, allows client nodes to communicate with server nodes multiple hops away by turning off the hop count decrementor. Also enables serverless WANs.
Note:	Banyan does not recommend using Serverless Networks on a WAN because the high cost increases delays and may terminate sessions. Wellfleet, however, does support such a configuration.
Instructions:	Enable only if this router also connects to a serverless network segment, and this is the inbound circuit toward the server. See the section entitled *Configuring the VINES Router on a Serverless Network Segment* for more information.

Parameter :	**Interface Cost**
Wellfleet Default:	0
Options:	0 or any cost
Function:	Overrides the default Banyan specified cost.
Instructions:	Select 0 to use the default Banyan specified cost. Otherwise, enter a new cost. For example, enter 3 to override the default Banyan cost and set the cost to 3.

Deleting VINES from the Wellfleet Router

You can delete VINES routing protocol from all Wellfleet router circuits on which it is currently enabled in two steps.

You begin from the Wellfleet Configuration Manager window and proceed as follows:

1. Select the Protocols/Vines/Delete Vines option.

   A window pops up and prompts "Do you REALLY want to delete Vines?".

2. Select OK.

   You are returned the Wellfleet Configuration Manager window. Vines is no longer configured on the Wellfleet router.

If you examine the Wellfleet Configuration Manager window, you will see that the connectors for circuits on which VINES was the *only* protocol enabled are no longer highlighted. Circuits must be reconfigured for these connectors; see chapter 3, *Configuring Circuits* for instructions.

Chapter 15

Configuring XNS

List of Figures

List of Tables

Configuring XNS

About this Chapter

This chapter describes how to configure the Wellfleet implementation of the Xerox Network Systems (XNS) router software. The section *XNS Overview* identifies the services provided by the XNS router. The *Implementation Notes* section provides guidelines you should follow if you are configuring XNS without RIP, or XNS on a Token Ring or SMDS interface. The *Editing XNS Parameters* section describes how to use the Configuration Manager to edit the XNS parameters.

Overview

Wellfleet's implementation of XNS is based on the *Xerox System Integration Standard* specification (Xerox Corporation, December, 1981), which is commonly referred to as *The Gray Book*.

The following sections provide a brief description of the XNS protocol stack and a description of the internetworking services pertinent to the Wellfleet XNS router.

XNS Protocol Stack

XNS was developed at the Xerox Palo Alto Research Center (PARC). Its layered architecture is a predecessor of the OSI architectural model. Both architectures are functionally similar. Figure 15-1 compares the OSI and XNS protocol stacks.

The layers that form XNS are as follows:

- Level 0 protocols handle the physical transmission of data between two points. Level 0 protocols are independent of XNS specifications. Instead, they depend on the transmission medium available between the two points engaged in communication. Examples of Level 0 protocols are Ethernet and Token Ring. Level 0 corresponds generally to Layers 1 and 2, the physical and data link layers, of the OSI model.
- The level 1 protocol, Internet Datagram Protocol (IDP), determines where each internet packet goes, addresses the source and destination of each internet packet, and selects the transmission medium. Level 1 corresponds generally to Layer 3, the network layer, of the OSI model.
- Level 2 protocols provide for the exchange of routing information between routers, handle the sequencing of packets within a packet stream, report transmission errors, retransmit packets in response to errors, suppress duplicate packets, and adjust the rate of packet transmission (flow control). Examples of Level 2 protocols are Routing Information Protocol (RIP), Error Protocol, and Echo Protocol. Level 2 corresponds to Layer 4, the transport layer, of the OSI model.

- Level 3 protocols are control protocols; they determine process interactions that involve remote resources, such as printer and file requests, and data structuring conventions. Level 3 corresponds generally to Layers 5 and 6, the session and presentation layers, of the OSI model.
- Level 4 protocols are application protocols that are implemented for specific platforms. Level 4 corresponds to Layer 7, the application layer, of the OSI model.

The sections describe the involvement of the Wellfleet XNS routing software in Levels 0, 1, and 2. Levels 3 and 4 do not involve routing, and are beyond the scope of this document.

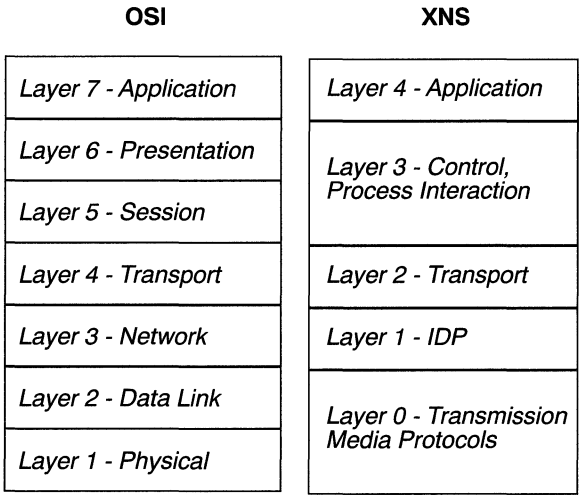


Figure 15-1. Comparison of OSI and XNS Protocol Stacks

Support for Level 0

Level 0 protocols handle the physical transmission of data between two points.

The Wellfleet XNS router supports the following Level 0 (physical and data link layer) protocols:

- Ethernet: Ethernet II
- Token Ring: SNAP
- FDDI: SNAP
- Frame Relay: Frame Relay SNAP
- SMDS: SMDS SNAP
- Point-to-Point (Wellfleet proprietary): Ethernet

Support for Level 1

Wellfleet implements Internet Datagram Protocol (IDP), the only XNS Level 1 protocol.

IDP determines where each internet packet goes, addresses the source and destination of each internet packet, and selects the transmission medium. It has the following characteristics:

- IDP is a connectionless datagram protocol. In other words, it does *not* need a channel established for delivery.
- IDP is unreliable. Higher level protocols assume the responsibility for reliability.

The Level 2 services provide IDP with the information necessary to route internet packets.

Support for Level 2

Level 2 protocols correspond to the Transport layers of the OSI model. The Wellfleet XNS router implements the following Level 2 protocols:

- Routing Information Protocol (RIP)
- Error Protocol
- Echo Protocol

The Wellfleet XNS router also provides the following Level 2 support:

- Routing of requests for a locally unavailable service to an available server on another XNS network. This support is referred to as External Server support.
- Static routing to other XNS networks
- Static routing to adjacent hosts

The sections that follow describe the support provided for these protocols and services.

Routing Information Protocol

Routing Information Protocol (RIP) provides workstations and routers with a means of exchanging information dynamically to establish the route with the fewest hops and shortest delay to each network.

Each XNS router maintains a RIP table. The RIP table contains the following information about every network in the XNS network topology:

- The network address of each network.
- The number of hops (cost) to that network.
- The address of the next hop node to which packets destined for that network will be forwarded.

Routers maintain RIP tables by exchanging request and response packets. Routers update their RIP tables with information from incoming response packets.

The header of each packet indicates the packet operation: request or response.

RIP request packets contain the number of the destination network in the header. A RIP request packet may be one of the following types:

- A general request broadcasted by a router to determine the fastest route to all networks on an internetwork. The value *FFFFFFFF* in the network number field within the RIP data indicates that the packet is a general request.
- A specific request broadcasted by a workstation or router to determine the fastest route to a particular network. One or more network numbers in the network number field within the RIP data indicates that the packet is a specific request.

Routers at the destination network issue RIP response packets. RIP response packets contain the network number and the number of hops and ticks required to get to the network. A RIP response may be one of the following types:

- A response to a request.
- An informational broadcast from a router issued every 30 seconds.
- An informational broadcast when a change occurs in the routing table. Examples of changes in the routing table are changes in cost information, changes to routes, aging of routes, and additions of routes to networks new to the table.
- An informational broadcast when an interface performs an orderly shutdown procedure or initializes.

To limit traffic, RIP broadcasts are limited to a router's immediate segments and are *not* forwarded by receiving routers.

Warning: The XNS router learns WAN addresses from RIP broadcasts received over WANs. The router stores XNS address/WAN address pairs for future use as next hop destinations. If RIP is not configured for a WAN interface, you must configure adjacent hosts for all transmission paths to nodes adjacent to Frame Relay or SMDS circuits when you configure an XNS interface. You must then configure static routes from the adjacent hosts to the next hop routers.

The XNS router allows you to enable RIP listen and RIP Supply functions for each XNS and/or XNS interface. When the Listen function is enabled, the XNS router adds routes received in RIP updates from neighboring routers to its own internal routing table. When the Supply function is enabled, the XNS router transmits RIP updates to routers on neighboring networks.

Error Protocol

The Error Protocol is an optional Level 2 protocol. It is intended to provide diagnostic and performance information.

The destination host detecting an error returns an Error Protocol packet to the socket of the host that generated the offending packet. The Error Protocol packet contains a copy of the first 42 bytes of the offending packet so that it can be validated by the source. The Packet Type field of the Error Protocol packet identifies the error number. Table 15-1 lists the XNS standard Error Protocol numbers. Wellfleet XNS routers report errors they detect using this standard.

The host that detected the error drops the offending packet after copying its first 42 bytes to the Data field of the error protocol packet.

Because the protocol is optional, the host receiving the Error Protocol packet may or may not use the information before dropping the packet. The Wellfleet XNS router does not use the information in the Error Protocol packets it receives.

Table 15-1. XNS Error Protocol Numbers

Error Number (Octal)	Description of Error
0	An unspecified error is detected at the destination.
1	A serious inconsistency, such as an incorrect checksum, is detected at the destination.
2	The destination socket specified in the offending packet does not exist in the destination host.
3	The destination dropped the packet because of resource limitations.
1000	An unspecified error occurred before reaching the destination.
1001	A serious inconsistency, such as an incorrect checksum, occurred before reaching the destination.
1002	The destination host cannot be reached from here.
1003	The packet's hop count reached its upper-bound threshold without reaching its destination.
1004	The packet is too large for an intermediate network. The Error Parameter field of the Error Protocol packet contains the maximum packet length allowed.

Echo Protocol

The Echo Protocol is a Level 2 protocol. It provides the following:

- A simple means to verify the existence and correct operation of a host's implementation of IDP.
- A simple means to verify the existence and correct operation of a path to such a host.

The Echo Protocol packet contains an Operation field, which indicates whether the packet is a request (1) or a response (2). The Wellfleet XNS router generates responses only to echo requests it receives on the well-known error socket, Socket 2. It does not generate echo request packets.

When the destination host receives an echo request packet, it generates a response packet and copies the data from the Data field of the request packet to the Data field of the response packet. The destination host then forwards the response packet to the source socket of the host that sent the echo request. This provides an opportunity for the requesting host to verify the data.

External Server

The Wellfleet XNS routing software features external server support. External server support is intended to provide client access to a service on another network if the service is not available on the client's network.

You enable external server support from the XNS Interface Parameters Window. Refer to the section *Editing XNS Interface parameters* for instructions.

When you enable external server support on a particular XNS interface, you specify the service request type to be routed and the destination of the service. The router then forwards incoming requests for that service type to the remote destination.

Note: You should enable external server support only when a service is not available on the local network. The default setting for this feature is Disabled.

Static Route

Static routes are manually configured routes that specify the next hop in the transmission path a datagram must follow based on the datagram's destination address. A static route specifies a transmission path to another *network*.

The Wellfleet XNS router allows you to configure static routes on each logical XNS interface. For example, in Figure 15-2 the route from the interface on Wellfleet Router Host ID 1 to Network 5 is a static route.

Static route support for XNS allows you to do the following:

- Direct all XNS traffic destined to a given network to an adjacent host.
- Reduce routing traffic by disabling RIP Supply on all or a subset of attached interfaces and manually configuring static routes.
- Eliminate all dynamic routing capabilities and all RIP supply and listen activities over an XNS interface.

Note: Unlike routes learned through RIP, static routes remain in the RIP tables until you delete them.

Warning: To establish a Data Link layer connection in a Frame Relay or SMDS network, which allows the router to send packets over a static route, you must configure an adjacent host, and edit the DLCI parameter in the XNS Adjacent Host Parameters Window.

Refer to the section *Editing Static Route Parameters* for detailed instructions and parameter definitions.

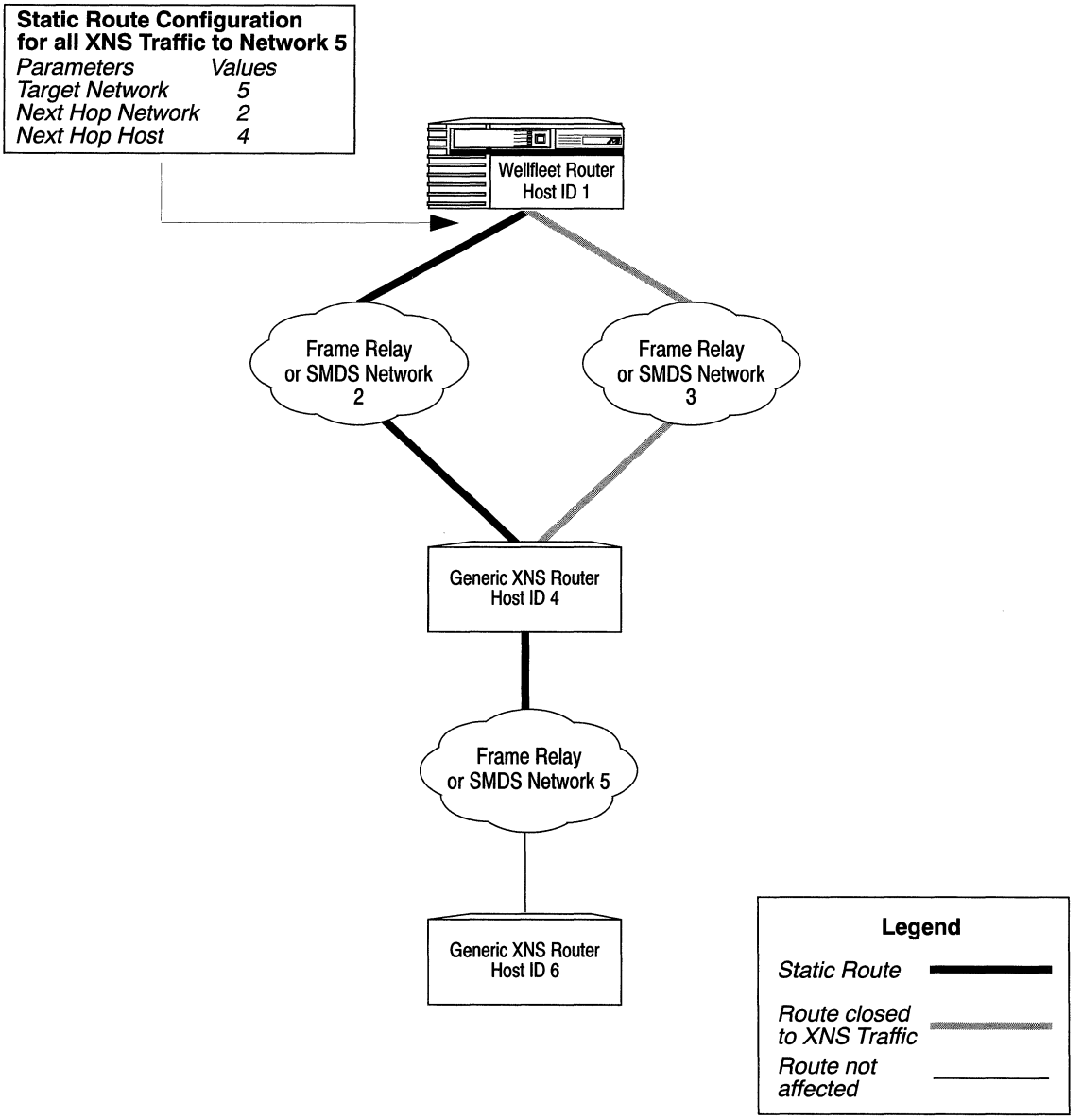


Figure 15-2. Static Route in a Sample Network

Adjacent Host

An adjacent host is a *network device* (that may or may not be a router) that is local to a directly connected network. For example, host 4 in Figure 15-3 is an adjacent host to Wellfleet Router Host ID 1. Host 6 is *not* an adjacent host because it is *not* connected logically to a directly adjacent network.

The Wellfleet XNS router allows you to specify static transmission paths to adjacent hosts. A static transmission path to an adjacent host establishes the data link connection necessary for packet transmission along a static route in a Frame Relay or SMDS network when RIP is not enabled. For example, in Figure 15-3 the XNS interface on Wellfleet Router Host ID 1 has host 4 configured as a statically adjacent host. This provides a data link connection that allows the static routing to occur between Host ID 1 and Network 5 in Figure 15-2.

With adjacent host support, you can do the following:

- You can configure the XNS router to map XNS addresses of network devices that are local to adjacent WANs to their associated WAN addresses.
- You can configure many static routes that use a single adjacent host as their next hop node, thereby reducing manual configuration tasks.

Note: You must use the DLCI (Data Link Control Identifier) parameter to identify a virtual circuit when you configure a static adjacent host in a Frame Relay network. You display this parameter by adding the adjacent host and then clicking the Edit button. Refer to the section *Editing Adjacent Host Parameters* for detailed instructions and parameter definitions.

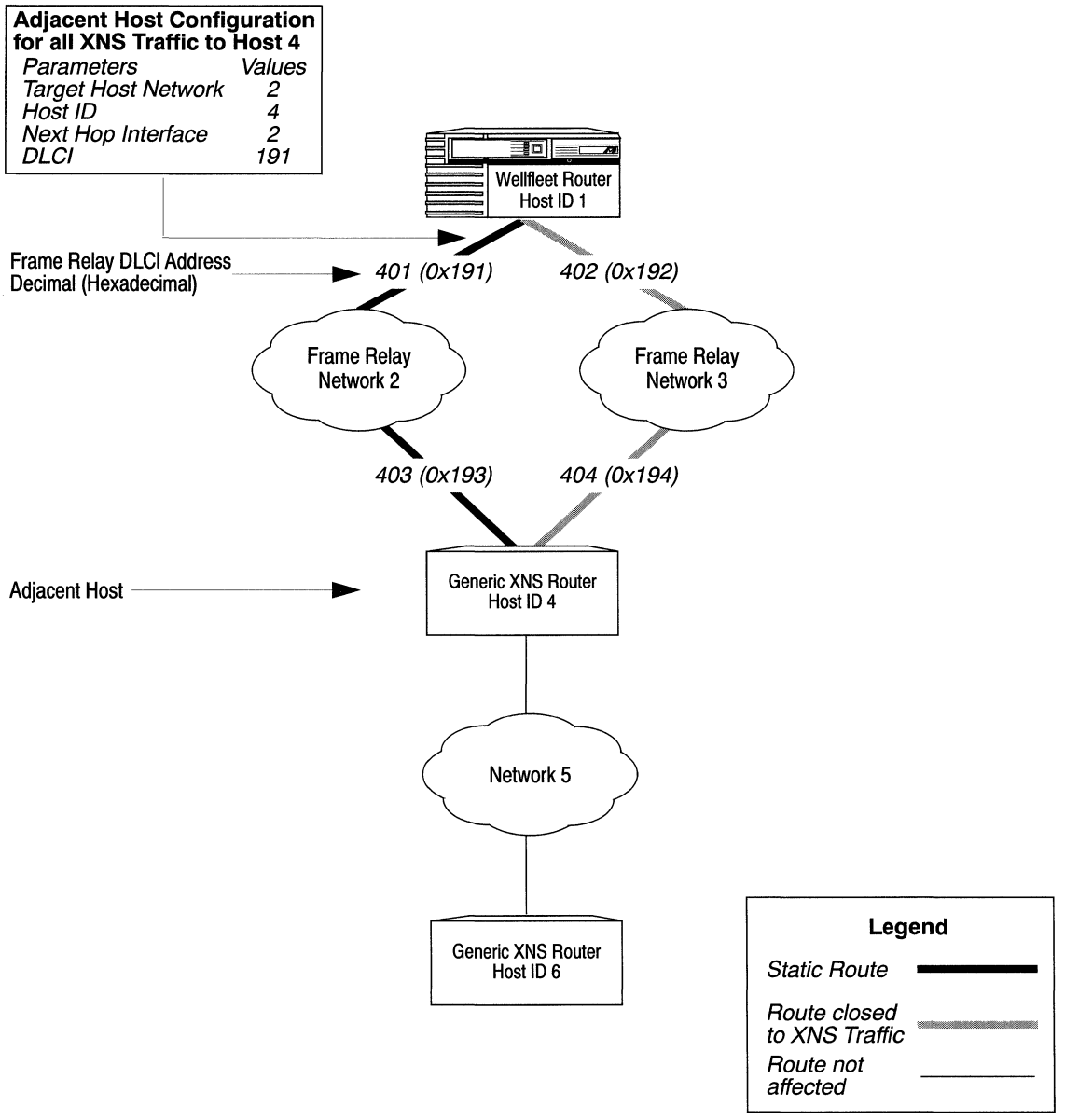


Figure 15-3. Static Adjacent Host in a Sample Network

XNS Bibliography

The following documents provided technical detail on XNS protocol implementation.

Xerox System Integration Standard (Xerox Corporation, December, 1981)

Xerox Network Systems Architecture General Information Manual (Xerox Corporation, 1985)

Implementation Notes

You should refer to the sections that follow only if you are configuring one of the following:

- XNS without RIP
- XNS on a Token Ring interface

Otherwise, refer to the section *Editing XNS Parameters*.

Configuring XNS without RIP

The XNS router learns WAN addresses from RIP broadcasts received over WANs. The router stores the XNS address/WAN address pairs in its RIP Table for future determination of next hop destinations.

Every XNS router on the internetwork learns about all of the other XNS routers through the propagation of RIP Tables. These tables can become very large in large internetworks. You may want to configure XNS without RIP to control the size of these tables and reduce bandwidth. However, you must do the following when you configure an XNS WAN interface without RIP:

1. Configure an adjacent host, and edit the DLCI parameter in the XNS Adjacent Host Parameters Window for each host on an adjacent Frame Relay or SMDS network.

   Refer to the section *Editing Adjacent Host Parameters* for detailed instructions.

2. Configure a static route to the next hop router for each adjacent host.

   Refer to the section *Editing Static Route Parameters* for detailed instructions.

Configuring a MAC Address on a Token Ring Interface

Any physical interface that can run in promiscuous mode, such as LANCE, ILACC, and FSI, allows multiple protocols to register a MAC address for which the protocol software can listen. Therefore, XNS can register its host number as the MAC address for each interface. However, if XNS is running over a Token Ring interface, you must enter the host ID in the MAC Address Override parameter and set the MAC Address Select parameter to Cnfg for *every* Token Ring interface on which XNS is running follows:

1. Select the Circuits/Edit Circuits option from the Configuration Manager Window.
2. Select the Token Ring circuit in the Circuit List Window and click the Edit button.
3. Select the Lines option in the Circuit Definition Window.
4. Select the interface from the Edit Lines Window and click the Edit button.
5. Enter the router's XNS host ID in the MAC Address Override parameter box.
6. Set the MAC Address Select parameter to Cnfg in the Token Ring Parameters Window.
7. Repeat steps b through e for every Token Ring circuit on which XNS is running.

Note: Refer to the chapter *Configuring Circuits* for more information about configuring circuits.

Editing XNS Parameters

As you configure the XNS router, you supply information that it uses to route packets through an XNS Internet. The instructions in the following sections describe how to edit XNS global and interface parameters. This section assumes you have configured an interface to support XNS. Refer to the chapter *Configuring Circuits* for instructions.

You use the Configuration Manager to edit XNS parameters. The configuration function you wish to perform determines the type of parameters you edit. Table 15-2 lists each configuration function and the corresponding section in this chapter.

Table 15-2. XNS Parameters and Configuration Functions

To Do the Following:	See this Section:
Enable or Disable XNS on the entire Wellfleet router.	*Editing XNS Global Parameters*
Reconfigure XNS on an interface.	*Editing XNS Interface Parameters*
Reconfigure the Routing Information Protocol (RIP) on an interface.	*Editing RIP Interface Parameters*
Add, edit, and delete adjacent hosts.	*Editing Adjacent Host Parameters*
Add, edit, and delete static routes.	*Editing Static Route Parameters*
Delete XNS from the entire Wellfleet router.	*Deleting XNS from the Wellfleet Router*

The sections that follow describe how to access and edit XNS parameters. The following information is provided for each parameter:

- Wellfleet default
- Valid options
- Parameter's function
- Instructions for setting the parameter

To edit the XNS parameters, you begin from the Wellfleet Configuration Manager Window, the first window displayed when you enter the Configuration Manager application (see Figure 15-4). Select the Protocols/XNS option. The XNS configuration options are displayed.

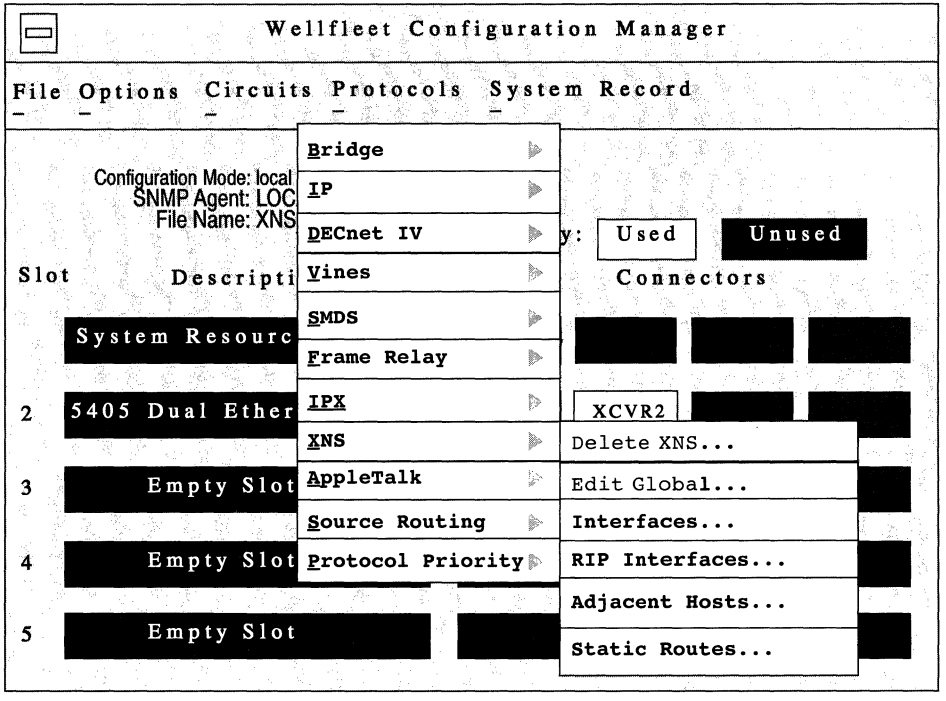


Figure 15-4. Wellfleet Configuration Manager Window

Editing XNS Global Parameters

To edit XNS Global parameters, begin at the Wellfleet Configuration Manager Window and proceed as follows:

1. Select the Protocols/XNS/Edit Global option.

   The Edit XNS Global Parameters Window appears (see Figure 15-5).

2. Edit those parameters you wish to change.
3. Click the Save button to save your changes and exit the window.

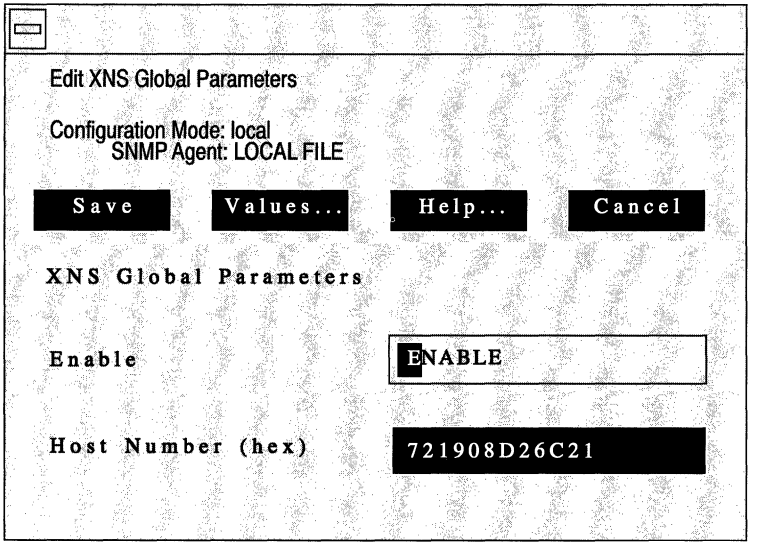


Figure 15-5. Edit XNS Global Parameters Window

A description of the parameters in this window follows.

Parameter: **Enable**

Wellfleet Default: The Configuration Manager automatically sets this parameter to Enable when you add XNS support to an interface.

Options: Enable/Disable

Function: Enables or disables XNS on the entire Wellfleet router.

Instructions: Select Enable if you have previously disabled the XNS router software and now wish to enable it.

Select Disable to disable the XNS router software.

Parameter: **Host Number**

Wellfleet Default: None. The setting displayed automatically in this box is the base host number you entered when adding XNS to a circuit for the first time.

Options: Any XNS host number.

Function: Sets the XNS host address and potential MAC address of the box.

Instructions: Use the setting displayed if it is the correct host number.

Enter the correct XNS host address in 12-digit hexadecimal notation.

Note: If the interface is on a Token Ring circuit, set the Token Ring MAC Address Select parameter to Cnfg. Refer to the section *Configuring a MAC Address on a Token Ring Interface* for more information.

Editing XNS Interface Parameters

When you added XNS to an interface, it took the XNS default settings. You can change these default settings by editing the XNS interface parameters.

To edit XNS interface parameters, begin at the Wellfleet Configuration Manager Window and proceed as follows:

1. Select the Protocols/XNS/Interfaces option to display the XNS Interfaces Window (see Figure 15-6).

   This window displays the network address in hexadecimal format of each interface you named when you added a circuit.

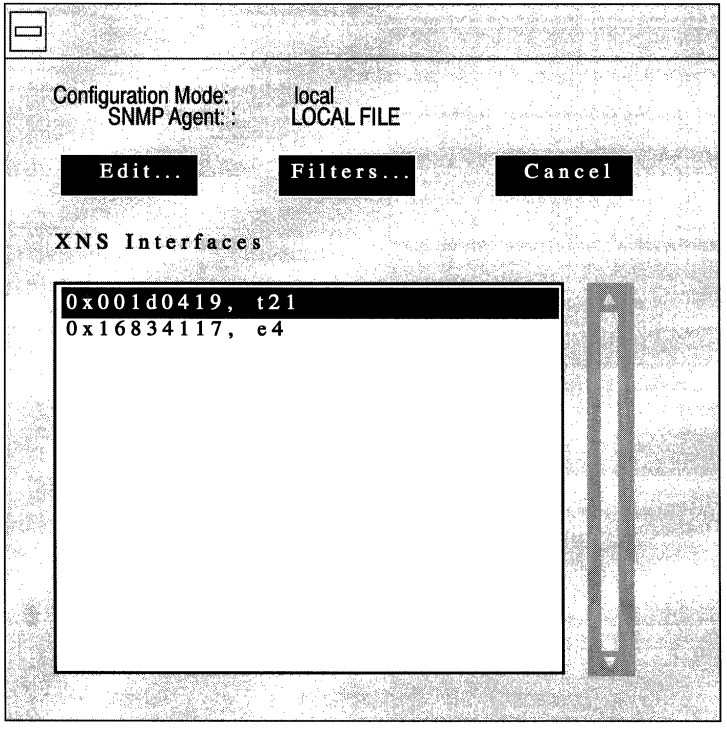


Figure 15-6. XNS Interfaces Window

2. Select the interface you wish to edit.
3. Click the Edit button to display the XNS Interface Parameters Window for that interface (see Figure 15-7).
4. Edit those parameters you wish to change.
5. Click the Save button to save your changes and exit the window.

Note: When you reconfigure an interface in dynamic mode, XNS restarts on that interface.

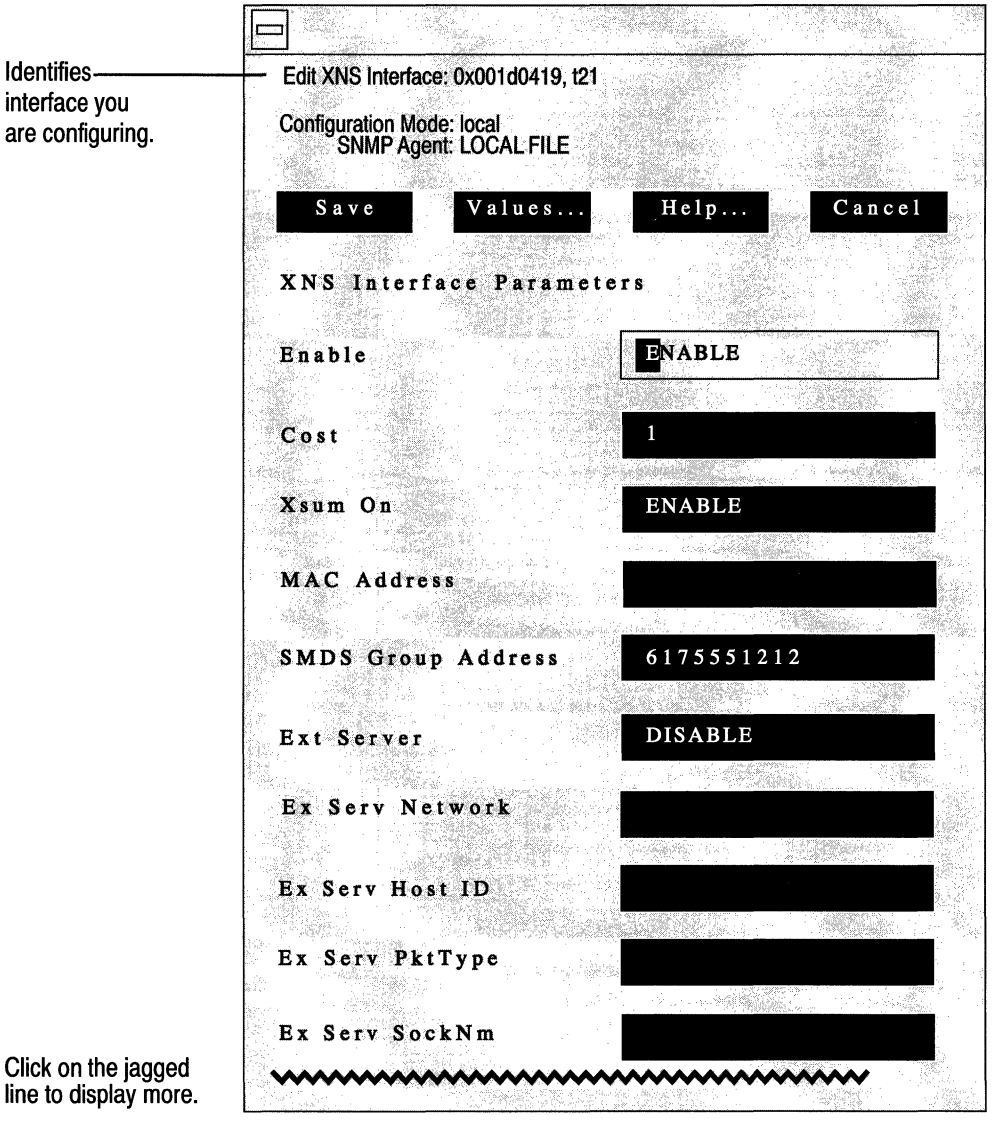


Figure 15-7. XNS Interface Parameters Window

A description of the parameters in this window follows.

Warning: When you reconfigure an interface in dynamic mode and select the Save button, XNS restarts on that interface.

Parameter :	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this interface-specific parameter to Enable when you add XNS support to this interface.
Options:	Enable/Disable
Function:	Enables or disables XNS routing on this interface.
Instructions:	Select Enable if you previously set this parameter to Disable and now wish the interface to support XNS routing. Select Disable only if you wish to disable XNS routing over this interface.

Parameter :	**Cost**
Wellfleet Default:	1 (for each hop)
Options:	0 to 15
Function:	Sets the cost (number of hops) for this interface. This parameter allows you to configure the shortest path. The cost is added to routes learned on this interface through RIP and is specified in subsequent RIP packets sent to other interfaces. XNS disposes of the packet when its hop count surpasses 15.
Instructions:	Enter the interface cost value. Standard RIP implementation assigns a cost of 1. Increasing this value causes the upper bound of 15 set by the RIP Network Diameter to be attained more rapidly.

Parameter :	**Xsum on**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Performs checksumming and compares the checksum to the number in the checksum field of each XNS packet. However, XNS does not perform a checksum on a packet it receives if the value of 0xffff is in the checksum field. XNS drops a packet if it performs a checksum on it and its value does not match the value in the checksum field.
Instructions:	Select Enable if you want XNS to perform checksumming. Select Disable to bypass checksumming.

Parameter :	**MAC Address**
Wellfleet Default:	None. The base host number you entered when adding XNS to the circuit overrides the MAC Address parameter.
Options:	Any valid MAC address
Function:	Specifies the MAC address of this interface
Instructions:	Leave this setting blank.

Parameter :	**SMDS Group Address**
Wellfleet Default:	None
Options:	The 10-digit North American Numbering Plan (NANP) telephone number for this interface that was provided by the common carrier.
Function:	Specifies the MAC group address of this interface in an SMDS network. This network-level interface parameter overrides the Group Address setting you entered when adding SMDS at the circuit level.
Instructions:	Leave blank if this interface is *not* on an SMDS circuit. Enter the 10-digit NANP telephone number for this interface that was provided by the common carrier. If only one telephone number is assigned to the circuit, enter the same telephone number as that entered in the Group Address box when you added SMDS to this circuit. You can display this number in the SMDS Interface Parameters Window. Refer to the chapter Configuring SMDS for more information.

Parameter :	**Ext Server**
Wellfleet Default:	Disable
Options:	Enable/Disable
Function:	Specifies whether external server capabilities are turned on. With this parameter enabled, the interface forwards packets of a particular type to a particular destination. The type is defined by the Ex Serv PktType parameter. The destination is determined by the remaining Ex Serv parameters on this screen.
Instructions:	Select Enable to turn on external server capabilities. Select Disable to turn off external server capabilities.

Parameter :	**Ex Serv Network**
Wellfleet Default:	Enable
Options:	Any valid network address
Function:	Specifies the network of the remote server to supply external server capabilities. This setting is used only if the Ext Server parameter is set to Enable.
Instructions:	Enter the network address of the remote server you want to supply external server capabilities. Leave blank if you are not using external server capabilities.

Parameter :	**Ex Serv Host ID**
Wellfleet Default:	0
Options:	Any valid host ID
Function:	Specifies the host ID of the remote server to supply external server capabilities. This setting is used only if the Ext Server parameter is set to Enable.
Instructions:	Enter the host ID of the remote server you want to supply external server capabilities. Leave blank if you are not using external server capabilities.

Parameter :	**Ex Serv PktType**
Wellfleet Default:	None
Options:	Any valid packet type
Function:	Specifies the packet type of the service requests to be forwarded to the remote server. This setting is used only if the Ext Server parameter is set to Enable.
Instructions:	Enter the packet type of the service requests to be forwarded to the remote server. Leave blank if you are not using external server capabilities.

Parameter :	**Ex Serv SockNM**
Wellfleet Default:	None
Options:	Any valid destination socket number.
Function:	Specifies the destination socket number of the remote server to which service requests are to be forwarded. This setting is used only if the Ext Server parameter is set to Enable.
Instructions:	Leave blank if you are not using external server capabilities, or leave blank if you are using external server capabilities and you want to forward all packets of the specified type that this interface receives to the specified remote server. Enter the destination socket number of the remote server to which service requests are to be forwarded.

Parameter :	**Frame Relay Broadcast**
Wellfleet Default:	ffffff (not displayed)
Options:	Default value or a user-specified Frame Relay broadcast address.
Function:	Specifies a Frame Relay broadcast address for this XNS interface.
Instructions:	Leave blank to accept the default value. With the default value, the XNS router sends all broadcast traffic through all logical connections associated with the XNS interface you are configuring. Enter a Frame Relay broadcast address to send all broadcast traffic through the XNS interface you are configuring.

Parameter :	**Frame Relay Multicast**
Wellfleet Default:	ffffff (not displayed)
Options:	Default value or a user-specified Frame Relay multicast address.
Function:	Specifies a Frame Relay multicast address for this XNS interface.
Instructions:	Leave blank to accept the default value. With the default value, the XNS router sends all multicast traffic through all logical connections associated with the XNS interface you are configuring. Enter a Frame Relay multicast address to send all multicast traffic through the XNS interface you are configuring.

Editing RIP Interface Parameters

Once you enable RIP on an interface, you can edit that interface in the RIP Interface Parameters Window for that interface. You enable RIP when you add a circuit. For instructions on how to enable RIP on an interface, see the *Configuring Circuits* chapter.

To edit RIP interface parameters for an XNS interface, begin at the Wellfleet Configuration Manager Window and proceed as follows:

1. Select the Protocols/XNS/RIP Interfaces option.

   The XNS RIP Interfaces Window appears (see Figure 15-8). This window displays the network address of each interface you named when you added a circuit.

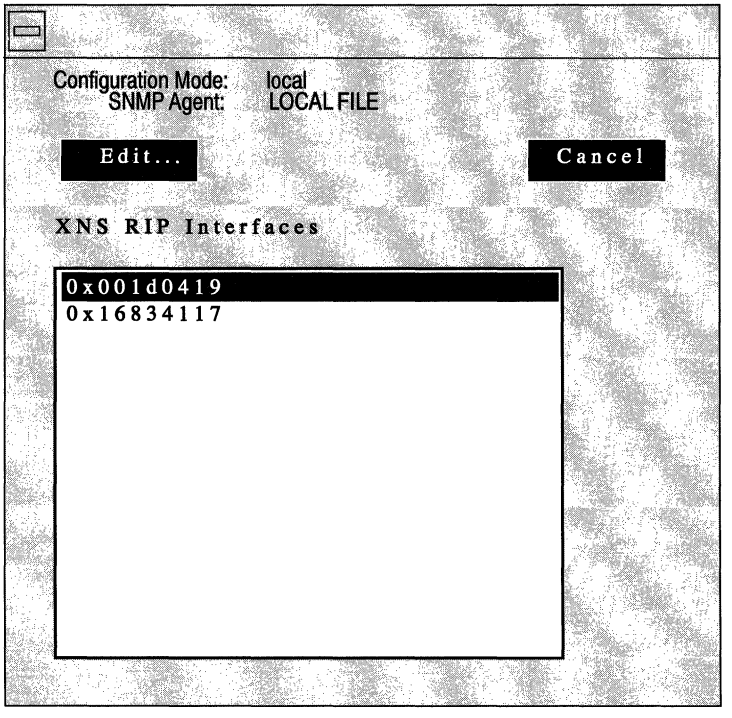


Figure 15-8. XNS RIP Interfaces Window

2. Select the interface you wish to edit.
3. Click on the Edit button.

   The XNS RIP Interface Parameters Window appears (see Figure 15-9).
4. Edit those parameters you wish to change.
5. Click the Save button to save your changes and exit the window.

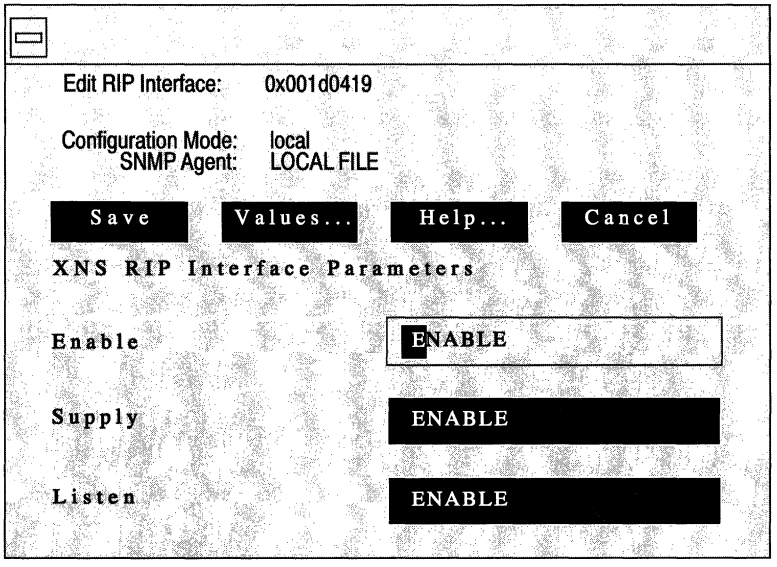


Figure 15-9. XNS RIP Interface Parameters Window

A description of the parameters in this window follows.

Parameter:	**Enable**
Wellfleet Default:	If you enabled RIP when you added the circuit or if you edited this circuit to support RIP, the Configuration Manager automatically sets this interface-specific RIP Enable parameter to Enable; otherwise, it is set to Disable.
Options:	Enable/Disable
Function:	Specifies whether the Routing Information Protocol (RIP) is enabled on this interface.
Instructions:	Select Enable to enable RIP on this interface. Select Disable to disable RIP on this interface.

Parameter:	**Supply**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether the interface transmits all RIP updates to routers in neighboring networks.
Instructions:	Select Enable to configure the interface to transmit all RIP updates. Select Disable to prohibit the interface from transmitting all RIP updates.

Parameter:	**Listen**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Specifies whether this interface listens to RIP updates from neighboring networks.
Instructions:	Select Enable to configure this interface to listen to RIP updates, and, thus, add received routing information to its internal routing table. Select Disable to configure the interface to ignore RIP updates from neighboring routers. Thus, the interface does not add received routing information to its internal routing table.

Note: If this parameter is set to Enable, a route filter can still prohibit the interface from updating its internal routing tables.

Editing Adjacent Host Parameters

The sections that follow describe how to add, edit, and delete adjacent host routes. You perform these functions from the XNS Adjacent Hosts Window (see Figure 15-10). Begin at the Wellfleet Configuration Manager Window and select the Protocols/XNS/Adjacent Hosts option. The XNS Adjacent Hosts Window appears.

Refer to the following sections to add, edit, and delete adjacent host routes.

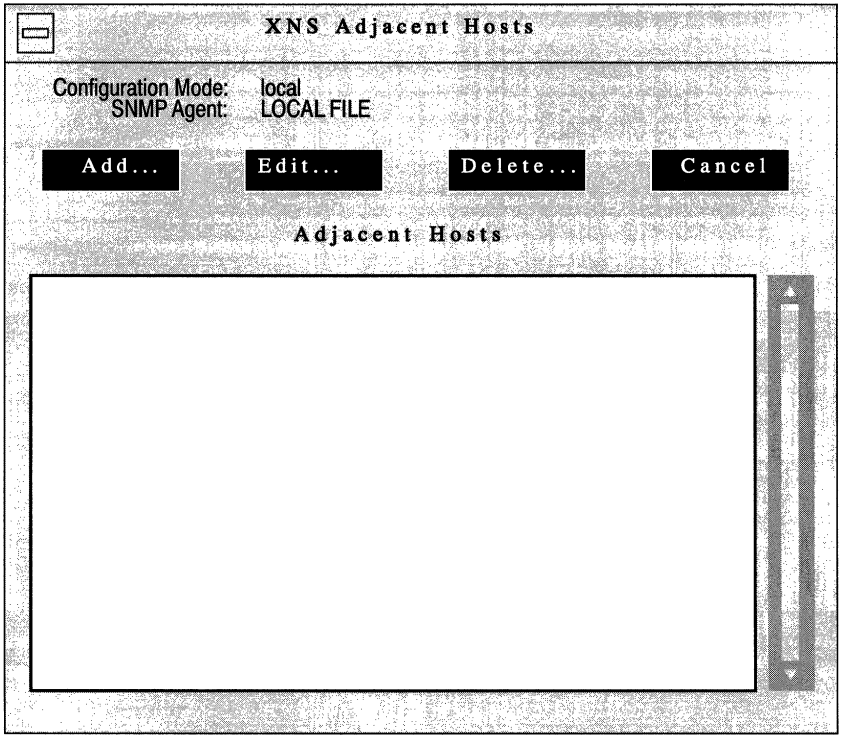


Figure 15-10. XNS Adjacent Hosts Window

Adding an Adjacent Host

To add an adjacent host, begin at the XNS Adjacent Hosts Window (see Figure 15-10) and proceed as follows:

1. Click on the Add button.

   The Add Adjacent Host Window appears (see Figure 15-11). This window contains the parameters required to add an adjacent host.

2. Edit those parameters you wish to change.

3. Click on the Add Adjacent Host button.

   The XNS Adjacent Host Parameters Window appears (see Figure 15-12). The next section describes how to edit the parameters in the XNS Adjacent Host Parameters Window.

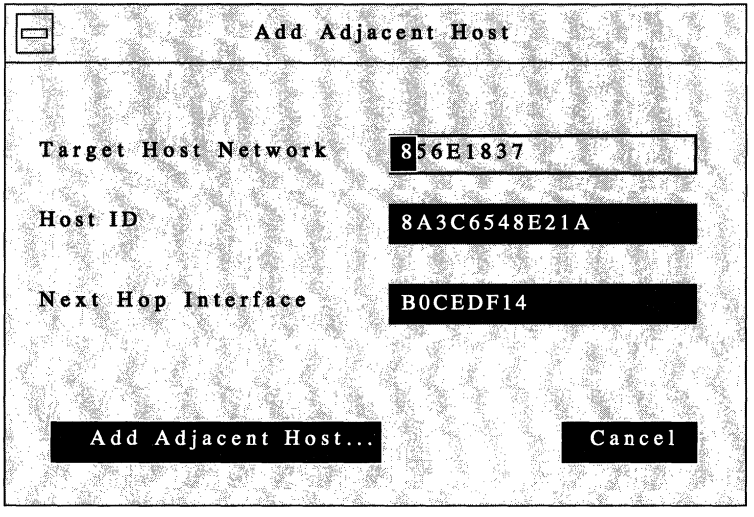


Figure 15-11. Add Adjacent Host Window

The parameters in the Add Adjacent Host Window are as follows.

Parameter :	**Target Host Network**
Wellfleet Default:	None
Options:	Valid network address of the static adjacent host.
Function:	Specifies the network address of the static adjacent host.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Host ID**
Wellfleet Default:	None
Options:	Valid host ID of the static adjacent host.
Function:	Specifies the host ID of the device for which you wish to configure an adjacent host.
Instructions:	Enter a host ID of up to 12 hexadecimal characters.

Parameter :	**Next Hop Interface**
Wellfleet Default:	None
Options:	Configured network address of the next hop.
Function:	Specifies the network address of the next hop.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Editing an Adjacent Host

You edit an adjacent host to change the settings of configurable adjacent host parameters, including the default settings.

The Configuration Manager does not allow you to change the Target Host Network and Host ID parameters. If you wish to change these parameters, you must delete the adjacent host and configure a new adjacent host. However, you can reconfigure all other parameters associated with an adjacent host.

The XNS Adjacent Host Parameters Window (see Figure 15-12) appears automatically when you follow the procedure to add an adjacent host as described in the previous section. To edit an existing adjacent host, begin at the XNS Adjacent Hosts Window (see Figure 15-10) and proceed as follows:

1. Select the adjacent host you wish to edit.
2. Click on the Edit button.

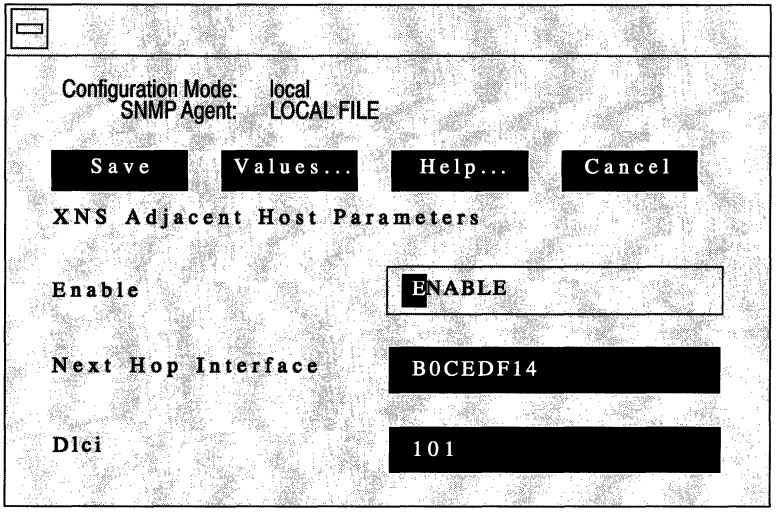


Figure 15-12. XNS Adjacent Host Parameters Window

When the XNS Adjacent Host Parameters Window is displayed, proceed as follows:

1. Edit those parameters you wish to change.
2. Click the Save button to save your changes and exit the window.

The parameters in this window are as follows.

Parameter :	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this parameter to Enable when you click on the Add Adjacent Host button in the Add Adjacent Host Window.
Options:	Enable/Disable
Function:	Specifies the state (active or inactive) of the adjacent host in the XNS routing tables.
Instructions:	Select Disable to make the adjacent host record inactive in the XNS routing table; the XNS router will not consider this adjacent host. Select Enable to make the adjacent host record active again in the XNS routing table.

Parameter :	**Next Hop Interface**
Wellfleet Default:	None
Options:	Configured network address of the next hop.
Function:	Specifies the network address of the next hop.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Dlci**
Wellfleet Default:	None
Options:	Data Link Control Identifier
Function:	Identifies the virtual circuit in a Frame Relay or SMDS network.
Instructions:	Enter a DLCI of up to 16 hexadecimal characters if the interface is on a Frame Relay or SMDS network. Leave blank if the interface is *not* on a Frame Relay or SMDS network.

Warning: The router cannot pass traffic through an interface to an adjacent host on a Frame Relay or SMDS network if the adjacent host is configured without the correct DLCI.

Deleting an Adjacent Host

To delete an adjacent host, select the adjacent host you wish to delete in the XNS Adjacent Hosts Window, and click on the Delete button (see Figure 15-10). The Delete XNS Adjacent Host Window appears. Click on the Delete button to delete the adjacent host.

Editing Static Route Parameters

XNS static routes are user-specified transmission paths that XNS internet packets follow. You configure static routes when you want to restrict the paths that packets can follow. Static routes, like routes learned through RIP, are maintained in the XNS routing table. Unlike routes learned through RIP, however, static routes do not time out. Static routes remain in the XNS routing table until they are reconfigured manually.

The sections that follow describe how to add, edit, and delete XNS static routes. You perform these functions from the XNS Static Routes Window (see Figure 15-13). Begin at the Wellfleet Configuration Manager Window and select the Protocols/XNS/Static Routes option. The XNS Static Routes Window appears.

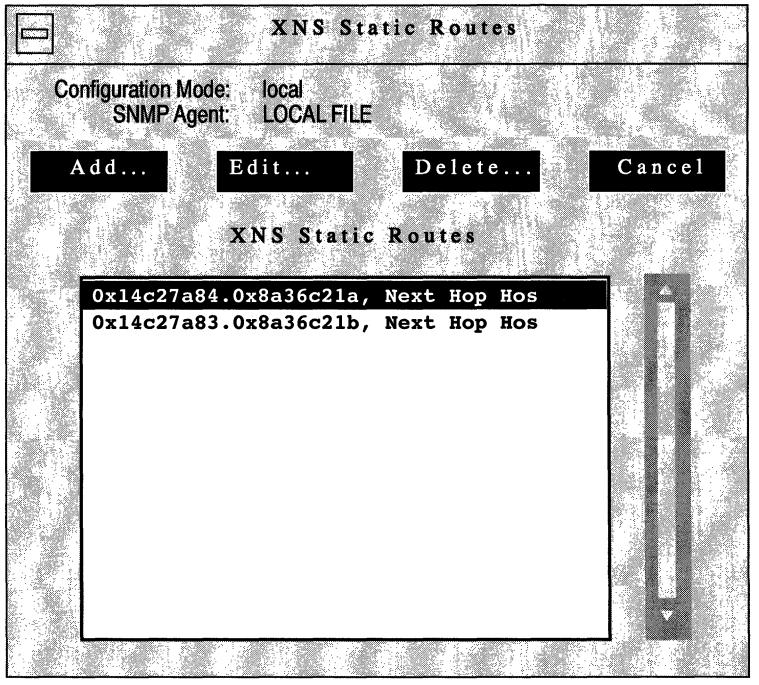


Figure 15-13. XNS Static Routes Window

Refer to the following sections to add, edit, and delete static routes.

Warning: To establish a Data Link layer connection in a Frame Relay or SMDS network, which allows the router to send packets over a static route, you must configure an adjacent host, and edit the DLCI parameter in the XNS Adjacent Host Parameters Window.

Adding a Static Route

To add a static route, begin at the XNS Static Routes Window (see Figure 15-13) and proceed as follows:

1. Click on the Add button.

   The XNS Add Static Route Window appears (see Figure 15-14). This window contains the parameters required to add a static route. (Static route parameters with default values are described in the next section.)

2. Edit those parameters you wish to change.

3. Click on the Add Static Route button.

   The XNS Static Route Parameters Window appears (see Figure 15-15). The next section describes how to edit the parameters in the XNS Static Route Parameters Window.

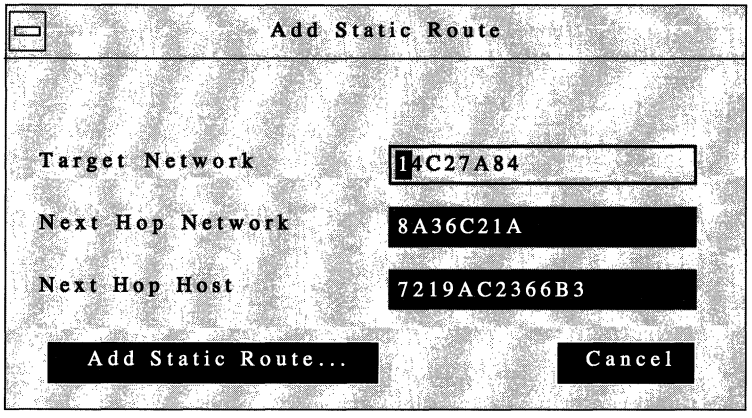


Figure 15-14. XNS Add Static Route Window

The parameters in the XNS Add Static Route Window are as follows.

Parameter :	**Target Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the address of the network to which you wish to configure the static route.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Next Hop Network**
Wellfleet Default:	None
Options:	Any valid network address in hexadecimal notation.
Function:	Specifies the network address of the next hop.
Instructions:	Enter a network address of up to eight hexadecimal characters.

Parameter :	**Next Hop Host**
Wellfleet Default:	None
Options:	Any valid host address in hexadecimal notation.
Function:	Specifies the address of the host to which you wish to configure the static route.
Instructions:	Enter a host address of up to 12 hexadecimal characters.

Editing a Static Route

You edit a static route to change the settings of configurable static route parameters, including the default settings.

The Configuration Manager does not allow you to reconfigure the Target Network and Next Hop Network parameters for a static route. If you wish to change these parameters, you must delete the static route and add a new route with the proper information. However, you can reconfigure all other parameters associated with a static route.

The XNS Static Route Parameters Window (see Figure 15-15) appears automatically when you follow the procedure to add a static route as described in the previous section. To edit an existing static route, begin at the XNS Static Routes Window (see Figure 15-13) and proceed as follows:

1. Select the static route you wish to edit.
2. Click on the Edit button.

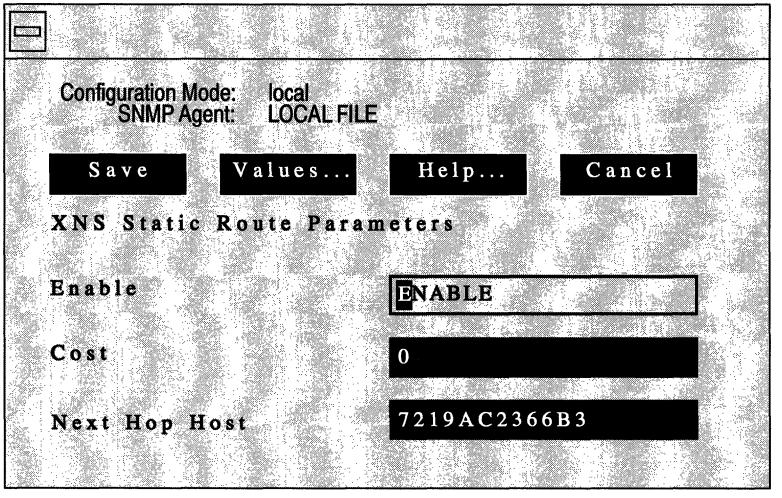


Figure 15-15. XNS Static Route Parameters Window

When the XNS Static Route Parameters Window is displayed, proceed as follows:

1. Edit those parameters you wish to change.
2. Click the Save button to save your changes and exit the window.

The parameters in the XNS Static Route Parameters window are as follows.

Parameter :	**Enable**
Wellfleet Default:	The Configuration Manager automatically sets this parameter to Enable when you click on the Add Static Route button in the Add XNS Static Route Window.
Options:	Enable/Disable
Function:	Specifies the state (active or inactive) of the static route record in the XNS routing tables.
Instructions:	Select Disable to make the static route record inactive in the XNS routing table; the XNS router will not consider this static route. Select Enable to make the static route record active in the XNS routing table.

Parameter:	**Cost**
Wellfleet Default:	0
Options:	0 to 15
Function:	Specifies the number of router hops added to an XNS data packet. The XNS router uses Cost when determining the best route for a datagram to follow. The Cost is also propagated through RIP. The default setting of 0 for static routes gives them priority over RIP-learned routes.
Instructions:	Enter the number of router hops.

Parameter:	**Next Hop Host**
Wellfleet Default:	None
Options:	Any valid host address in hexadecimal notation
Function:	Specifies the address of the host to which you wish to configure the static route.
Instructions:	Enter a host address of up to 12 hexadecimal characters.

Deleting a Static Route

To delete a static route, first select the static route you wish to delete in the XNS Static Routes Window (see Figure 15-13). Then click on the Delete button to display the Delete XNS Static Route Window. Click on the Delete button to delete the static route.

Deleting XNS from the Wellfleet Router

You can delete XNS from all Wellfleet router interfaces on which it is currently enabled in two steps.

To delete XNS (if it is enabled), begin at the Wellfleet Configuration Manager Window and complete the following steps:

1. Select the Protocols/XNS/Delete XNS option.

   A confirmation window appears.

2. Select OK.

   You are returned to the Wellfleet Configuration Manager window. XNS is no longer configured on the Wellfleet router.

Note: If you deleted XNS, the connectors for those interfaces on which the XNS was the *only* protocol enabled are no longer highlighted in the Wellfleet Configuration Manager Window. Interfaces must be reconfigured for these connectors; see *Configuring Circuits* for instructions.

Chapter 16

Configuring Filters

List of Figures

List of Tables

Configuring Filters

About this Chapter

This chapter provides the following:

- An overview of traffic filtering
- A description of templates
- A description of the fields, field ranges, and actions specific to each protocol
- An explanation of how to specify user-defined fields
- An explanation of how to use the Configuration Manager to configure filters, which includes:
  - Adding a filter to an interface
  - Creating a template from scratch
  - Copying a template
  - Editing a template (its fields, ranges, and actions)
  - Deleting a template
  - Editing a filter (its fields, ranges, and actions)
  - Deleting a filter

You should read this chapter if you are responsible for configuring traffic filters for your network. If you are already familiar with Version 7 traffic filters, and with the fields and actions associated with the protocol(s) for which you wish to create filters, you can go directly to *Using the Configuration Manager to Configure Filters*.

Note: This chapter provides instructions for creating Version 7 filters. With Version 7, you can create any filter that you previously may have created with Version 5. To do this, Wellfleet recommends you use the instructions in this chapter, as the procedure is easier. However, if you prefer, see Appendix C, *Converting Version 5 Traffic Filters*, which provides a manual conversion algorithm for this purpose.

Traffic Filters

Traffic filters enable the router to selectively relay or drop a packet, frame, or datagram based on standard protocol fields or user-defined fields. They apply to incoming traffic only, and are used primarily for security. For example, suppose a company wants only a certain few people to be able to access its financial network. A filter can be constructed denying everyone access to the financial network except for those certain few people.

All filters are created from templates (files that hold the filtering information), and consist of the following three components:

- Field

  A filtering field is a part of a frame, packet or datagram header that you specify to be examined on each incoming frame.

- Range

  A range of numeric values is associated with a filtering field.

- Action

  An action defines what happens to an incoming packet that matches a filter.

Filters are associated with a protocol, and further, with an interface. Traffic filters are supported in the following protocols:

- Bridge
- IP
- DECnet Phase IV
- VINES
- Source Routing
- IPX
- XNS

Each of these protocols allows up to 31 filters per interface. As filters are added to an interface, they are numbered chronologically in the following fashion: Rule 1, Rule 2, Rule 3, and so on.

Templates and Filters

It is important for you to understand the difference between a template and a filter. A template is simply a file that holds specific filtering information (fields, ranges, and actions). A filter is created when you apply (save) a template to an interface. You can apply a single template to as many interfaces as you want; see Figure 16-1. Once a template is created, it exists for future use unless you delete it.

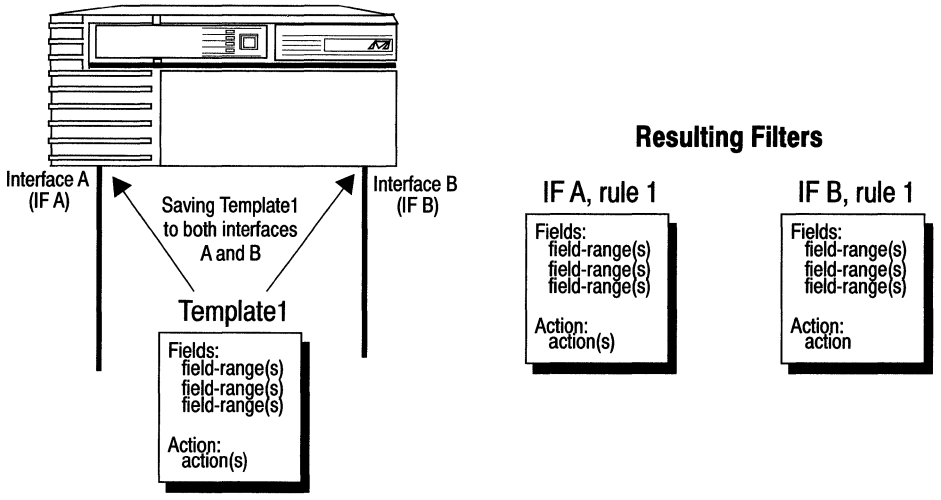


Figure 16-1. Using a Template to Create Filters

Generally, when you create a template, you first assign it a one-word name. It is a good idea to give your template a descriptive name. For example, if you are building a template that is going to contain filtering information instructing the interface to drop all DECnet Phase IV traffic with a Source Node value of 3, you should probably name it something like *decSnode3*.

Note: A template must contain fields and actions for *one protocol only* (Bridge, IP, DECnet Phase IV, VINES, IPX, XNS or Source Routing).

Once you have named your template, you select the fields and assign ranges for which each packet will be checked. You then select the action(s) that will be imposed on any packet that matches at least one range for every field in the filter. After you have specified this

information, you save it, thus creating a template. A more detailed, step-by-step example of creating a filter, which includes creating a template from scratch, is provided in *Adding a Filter to an Interface*.

When you want to add a filter to an interface, you have several options:

- If there is an existing filter on the interface that contains filtering instructions similar to what you want, you may edit this filter and save it (see *Editing a Filter*). The changes to the filter are valid only on this interface. The template originally used to create the filter does not change.
- If there is a template that contains the exact filtering instructions that you desire for this interface, you can apply (save) that template to this interface.
- If there is a template that contains filtering instructions similar to what you want, you can copy the template, rename, and edit it (see section entitled *Editing a Template*). When you save the changes, you have created a new template. You can now apply this template to any interface for which its filtering instructions are appropriate.
- If there is no template containing filtering instructions similar to what you want for this interface, you must create a template from scratch.

It is this last case that is discussed in *Adding a Filter to an Interface*.

Note: Because filters are created on a per protocol basis, you *must* become familiar with the specific fields and actions used for filtering by each protocol. They are described in the next section. If you are already familiar with them, go directly to *Using the Configuration Manager to Configure Filters*.

Filtering Fields, Ranges and Actions

As previously described, all filters are created from templates (files that hold filtering information), which consist of three components:

- Field

  A filtering field is a part of a packet, frame or datagram header that you specify to be checked on each incoming frame. These fields are protocol-specific. For example, in the bridge protocol, you can specify the MAC Source Address as a filtering field. This causes each incoming frame's MAC Source Address to be inspected. Each filtering field has one or more ranges associated with it.

- Range

  A range is associated with a filtering field. There must be at least one range per field. A range can consist of just one value, or it can be a set of values. You must specify a minimum and a maximum value for each range. For example, if you specify MAC Source Address as a filtering field, you must specify exactly which address(es) to filter. You could specify 0x0000A2000001 as the minimum value and 0x0000A2000003 as the maximum value. Then, all incoming packets would be checked to see if their MAC Source Address field was between 0x0000A2000001 and 0x0000A2000003, inclusive.

Note: When you enter values for minimum value and maximum value, the Configuration Manager assumes the value is a decimal number. If you want to enter a hexadecimal number, you *must* use the prefix 0x.

- Action

  An action defines what happens to an incoming packet that matches one of the ranges for every filtering field in the filter. Actions are protocol-specific except for the following two:

  — Drop

  Specifies that any frame that matches the filter will be discarded.

  — Log

  Specifies that for any frame that matches the filter, an event message will be recorded in the Event Log. The Log action can be combined with any other action; however, it should be used to record abnormal events only. Otherwise, the event log will fill up with filtering messages and thus become useless.

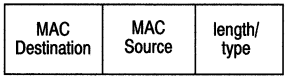
The following sections describe each protocol's predefined filtering fields and actions.

Bridge Fields and Actions

The bridge claims the most complex filtering scenario because so many encapsulations and media types are supported. That means that there are a lot more fields on which you can filter traffic. The bridge can be configured to filter frames based on the header fields within each of the four encapsulation methods supported by the bridge (see Figure 16-2). The bridge also supports user-defined filters. The supported encapsulation methods are:

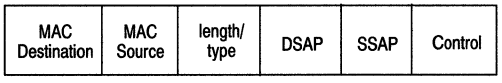
- Ethernet
- IEEE 802.2 logical link control
- IEEE 802.2 LLC with SNAP header
- Novell Proprietary

Ethernet Header



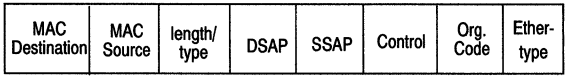
48-bit MAC destination address
48-bit MAC source address
16-bit length/type is TYPE (> 1518)

IEEE 802.2 LLC Header



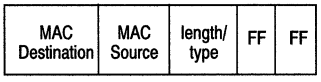
48-bit MAC destination address
48-bit MAC source address
16-bit length/type is LENGTH (<1519)
8-bit DSAP
8-bit SSAP
8-bit Control

IEEE 802.2 LLC w/SNAP Encapsulation



48-bit MAC destination address
48-bit MAC source address
16-bit length/type is LENGTH (<1519)
DSAP/SSAP/CTRL is 0xAAAA03
24-bit Organizational Code
16-bit Ethertype

Novell Proprietary Encapsulation



48-bit MAC destination address
48-bit MAC source address
16-bit length/type is LENGTH (<1519)
next 16 bits are all ones (part of IPX header

Figure 16-2. Headers of Encapsulation Methods Supported by the Bridge

Pre-Defined Fields

All frame headers include both a MAC Destination Address and a MAC Source Address field; therefore, filtering on these two fields is possible for all bridge-supported encapsulations. Aside from that, each encapsulation method has specific, pre-defined fields on which a frame can be filtered.

Table 16-1 shows the encapsulation support for each physical access medium, and Table 16-2 illustrates the pre-defined filtering fields for each encapsulation method.

Table 16-1. Bridge-Supported Encapsulation/Media Matrix

	Encapsulation Method			
	Ethernet	**802.2**	**SNAP**	**Novell**
Ethernet/802.3	Yes	Yes	Yes	Yes
FDDI	No	Yes	Yes	No
Point-to-Point	Yes	Yes	Yes	Yes
Token Ring	No	Yes	Yes	No

Note: There is no length/etype field in FDDI frames; there can be source routing information between the MAC Source Address and the DSAP field in a token ring frame.

Table 16-2. Pre-Defined Filter Fields for the Bridge

Encapsulation Method	Pre-Defined Fields
All	MAC Source Address MAC Destination Address
Ethernet	Ethernet type
802.2	Length SSAP DSAP Control
SNAP	Length Protocol ID/Organization code Ethertype

Note: There are no additional filtering fields for Novell; it allows filtering only on the MAC Source and MAC Destination Address fields.

User-Defined Fields

The Bridge supplements basic filtering functionality by providing the ability to filter traffic based upon specified bit pattern(s) contained within either the MAC or data-link header. When creating a filter on user-defined fields, you specify the Reference, Offset, and Length, that together describe the location of the field on the incoming packet.

- Reference

  Positions the filtered bit pattern within the incoming frame. For the Bridge there are two reference points: the first is at the beginning of the MAC header, and the second is at the beginning of the Data-Link header.

- Offset

  Positions the filtered bit pattern (measured in bits) within either the MAC-level or data-link-level header.

- Length

  Specifies the bit-length of the filtered field.

After specifying the Reference, Offset, and Length of your field, you specify one or more ranges for that field. For more information, see *Specifying User-Defined Fields* later in this chapter. The following tables show Reference, Offset, and Length for filtering fields supported by each encapsulation method

Table 16-3. Reference, Offset, and Length of Common Bridge Fields

Field	Reference	Offset	Length
MAC Destination Address	MAC	0	48
MAC Source Address	MAC	48	48

Table 16-4. Reference, Offset, and Length of Ethernet Encapsulation Fields

Field	Reference	Offset	Length
Ethernet type	MAC	96	16

Table 16-5. Reference, Offset, and Length of 802.2 Encapsulation Fields

Field	Reference	Offset	Length
Length	MAC	96	16
DSAP	DATA_LINK	0	8
SSAP	DATA_LINK	8	8
Control	DATA_LINK	16	8

Table 16-6. Reference, Offset, and Length of SNAP Encapsulation Fields

Field	Reference	Offset	Length
Length	MAC	96	16
Protocol ID/Organization Code	DATA_LINK	24	24
Ethertype	DATA_LINK	48	16

Actions

Aside from the Drop and Log actions common to all the protocols, the Bridge also claims two Bridge-specific actions. They are:

- Flood

  Specifies that any frame that matches the filter will be forwarded onto all bridge circuits except for the circuit from which it was received.

- Forward to Circuit List

  Specifies that any frame that matches the filter will be forwarded to certain circuits that you specify.

Remember that the Log action can be combined with any of the actions. However, Log should be used only to record abnormal events; otherwise, the event log will fill up with filtering messages and thus become useless.

IP Fields and Actions

IP can be configured to filter frames based on fields within the IP header. IP also supports user-defined fields.

Pre-Defined Fields

IP pre-defined filtering fields include:

- Type of Service
- IP Destination Address
- IP Source Address
- UDP Source Port
- UDP Destination Port
- TCP Source Port
- TCP Destination Port
- Protocol

User-Defined Fields

IP supplements basic filtering functionality by providing the ability to filter IP traffic based upon specified bit pattern(s) contained within the IP header or the header of the upper level protocol (TCP or UDP, for example) conveyed within the IP datagram. When creating a filter on user-defined fields, you must specify Reference, Offset, and Length, that together describe the location of the filter field on an incoming packet.

- Reference

  Positions the filtered bit pattern within the incoming frame. There are two reference points for IP. The first is Header Start, which is the beginning of the IP header. The second is Header End, which is actually the beginning of the UDP or TCP header.

- Offset

  Positions the filtered bit pattern (measured in bits) within either the IP, or higher level protocol header.

- Length

  Specifies the bit-length of the filtered field.

After specifying the Reference, Offset, and Length of your field, you specify one or more ranges for that field. For more information, see *Specifying User-Defined Fields* later in this chapter. Table 16-7 shows the Reference, Offset, and Length of each IP filtering field.

Table 16-7. Reference, Offset, and Length of IP Filtering Fields

Field	Reference	Offset	Length
Type of Service	HEADER_START	8	8
Protocol	HEADER_START	72	8
Source IP Address	HEADER_START	96	32
Destination IP Address	HEADER_START	128	32
UDP/TCP Source Port	HEADER_END	0	16
UDP/TCP Destination Port	HEADER_END	16	16

Actions

Aside from the Drop and Log actions common to all the protocols, IP also claims two IP-specific actions. They are:

- Forward to Next Hop

  Specifies that any frame that matches the filter will be forwarded the next hop router. You are required to specify the IP address of the next hop router. If the next hop router is not reachable, any packets matching the filter will be forwarded normally, unless Drop If Next Hop is Down is also specified.

- Drop if Next Hop is Down

  Specifies that if the address specified in Forward to Next Hop is unreachable, the frame is dropped. Drop if Next Hop is Down is valid only when Forward to Next Hop is in use.

Remember that the Log action can be combined with any other action. However, Log should be used only to record abnormal events; otherwise, the event log will fill up with filtering messages and become useless.

DECnet Phase IV Fields and Actions

DECnet Phase IV claims a simple filtering scenario because filtering can be done only on pre-defined fields.

Pre-Defined Fields

DECnet Phase IV pre-defined filtering fields include:

- Destination Area
- Destination Node
- Source Area
- Source Node

Table 16-8 shows the Reference, Offset, and Length of each DECnet predefined filtering field.

Table 16-8. Reference, Offset, and Length of DECnet Filtering Fields

Field	Reference	Offset	Length
Destination Area	HEADER_START	10	6
Destination Node	HEADER_START	22	10
Source Area	HEADER_START	42	6
Source Node	HEADER_START	54	10

Actions

DECnet Phase IV filtering actions include only Drop and Log.

VINES Fields and Actions

VINES can be configured to filter frames based on fields within the VINES IP header. User-defined filters are not supported in VINES.

Pre-Defined Fields

VINES pre-defined filtering fields include:

- Protocol Type
- Destination Address
- Source Address

Table 16-9 shows the Reference, Offset, and Length of the VINES predefined filtering fields.

Table 16-9. Reference, Offset, and Length of VINES Filtering Fields

Field	Reference	Offset	Length
Protocol Type	HEADER_START	40	8
Destination Address	HEADER_START	48	48
Source Address	HEADER_START	96	48

Actions

VINES filtering actions include only Drop and Log.

IPX Fields and Actions

IPX can be configured to filter frames based on fields within the IPX IP header. IPX does not support user-defined filters.

Pre-Defined Fields

IPX pre-defined filtering fields include:

- Destination Network
- Source Network
- Destination Socket
- Source Socket
- Destination Address
- Source Address

Table 16-10 shows the Reference, Offset and Length of the IPX predefined filtering fields.

Table 16-10. Reference, Offset, and Length of IPX Filtering Fields

Field	Reference	Offset	Length
Destination Network	HEADER_START	48	32
Source Network	HEADER_START	80	48
Destination Socket	HEADER_START	128	16
Source Socket	HEADER_START	144	32
Destination Address	HEADER_START	180	48
Source Address	HEADER_START	228	16

Actions

IPX filtering actions include only Drop and Log.

XNS Fields and Actions

XNS can be configured to filter frames based on fields within the XNS IP header. XNS does not support user-defined filters.

Pre-Defined Fields

XNS pre-defined filtering fields include:

- Destination Network
- Source Network
- Destination Socket
- Source Socket
- Destination Address
- Source Address

Table 16-11 shows the Reference, Offset and Length of the XNS predefined filtering fields.

Table 16-11. Reference, Offset, and Length of XNS Filtering Fields

Field	Reference	Offset	Length
Destination Network	HEADER_START	48	32
Source Network	HEADER_START	80	48
Destination Socket	HEADER_START	128	16
Source Socket	HEADER_START	144	32
Destination Address	HEADER_START	180	48
Source Address	HEADER_START	228	16

Actions

XNS filtering actions include only Drop and Log.

Source Routing Fields and Actions

Source Routing can be configured to filter frames based on fields within the Source Routing header. User-defined filters are also supported in Source Routing.

Note: Because two distinctly different types of frames (Specifically Routed Frames and Explorer Frames), exist in Source Routing, care must be taken when you create Source Routing filters. Keep in mind that any filter you create affects *both* types of frames.

Pre-Defined Fields

Source Routing pre-defined filtering fields include:

- Next Ring
- Destination MAC Address
- Source MAC Address
- DSAP
- SSAP
- Destination NetBIOS Name
- Source NetBIOS Name

Note: The Source and Destination NetBIOS name must be entered as the ASCII equivalent of the first 15 characters of the name. You must use ASCII spaces (0x20) to pad names to 15 characters if the name is less than 15 characters.

User-Defined Fields

Source Routing supplements basic filtering functionality by providing the ability to filter traffic based upon specified bit pattern(s) contained within the Source Routing header. When creating a filter on user-defined fields, you specify the Reference, Offset, and Length, that together describe the location of the field on the incoming packet.

- Reference

  Positions the filtered bit pattern within the incoming frame. For Source Routing there are three reference points: Next Ring, Header Start and Data Link.

- Offset

  Positions the filtered bit pattern (measured in bits) within either Next Ring, or the MAC-level or data-link-level header.

- Length

  Specifies the bit-length of the filtered field.

After specifying the Reference, Offset, and Length of your field, you specify one or more ranges for that field. For more information, see *Specifying User-Defined Fields* later in this chapter. Table 16-12 shows the Reference, Offset, and Length for Source Routing filtering fields.

Table 16-12. Reference, Offset, and Length of Source Routing Filtering Fields

Field	Reference	Offset	Length
Next Ring	NEXT_RING	0	12
Destination MAC Address	HEADER_START	0	48
Source MAC Address	HEADER_START	48	48
DSAP	DATA_LINK	0	8
SSAP	DATA_LINK	8	8
Destination NetBIOS Name	DATA_LINK	120	120
Source NetBIOS Name	DATA_LINK	248	120

Note: If the filter you create includes Next Ring as a field, only Specifically Routed Frames will be affected. The Next Ring field does not have any meaning for Explorer Frames.

If a filter you create includes the MAC Address, it must be defined in canonical format. This is bit-swapped from the order seen on token ring networks. In addition, the swapped source address must have the 0x01 bit turned on to account for the RII bit, which indicates the presence of the Routing Information Field.

Actions

Aside from the Drop and Log actions common to all the protocols, Source Routing claims one additional Source Routing-specific action. It is:

- Direct IP Explorers

  Specifies that any *explorer* frame that matches the filter will be sent to some number of IP addresses. You are required to specify these IP addresses.

  IP encapsulation must be configured for this action to be valid. If it is not configured, and a frame matches the filter, it will be flooded as if no filter existed.

Specifying User-Defined Fields

When you use the Configuration Manager to create or edit a template (described in the next section), you may add or edit filtering fields. When you access the appropriate menu to add a field, each pre-defined filtering field is represented as an option in that menu. The one other option is the User-Defined option. The User-Defined option allows you to set up specialized filtering fields based on bit pattern(s) within a packet's header. User-defined filters are supported only by the Bridge and IP.

Setting up user-defined fields is similar to setting up pre-defined fields, except that you must specify the field's location within the packet. (With pre-defined fields, you do not have to do that; their locations are established.) So, essentially, there is one extra step (window) required to specifying a user-defined field.

When you select the User-Defined option, the User-Defined Filter Field Window appears. In this window, you specify the field's location within the header. To do this, you must set the field's Reference, Offset, and Length. Then, you specify a range associated with the bit field described by the Reference, Offset, and Length.

For example, suppose that you are bridging VINES traffic over Ethernet, and you want to drop all packets with a destination network number of 1234 (hex), you would set up filtering fields as follows:

1. Specify an Ethernet Type field of 0xBAD (VINES). Ethernet Type is a pre-defined field.
2. Determine the Reference, Offset, and Length values of the Destination Network field within the header (see Figure 16-3).

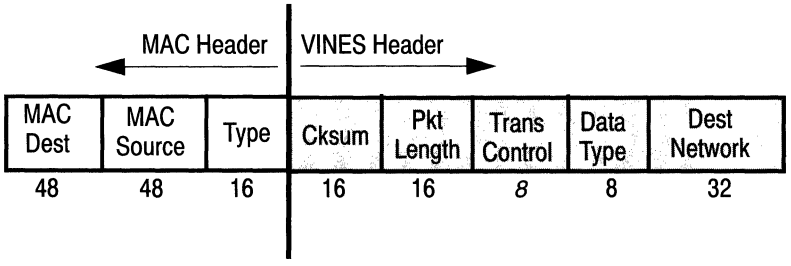


Figure 16-3. VINES Header

3. Set the Reference, Offset, and Length in the Add User-Defined Field Window, as follows (see Figure 16-4).
   - — Reference = MAC (beginning of frame)
   - — Offset = 160 bits (sum of all fields that precede the Destination Network field, or 48+48+16+16+16+8+8)
   - — Length = 32 bits

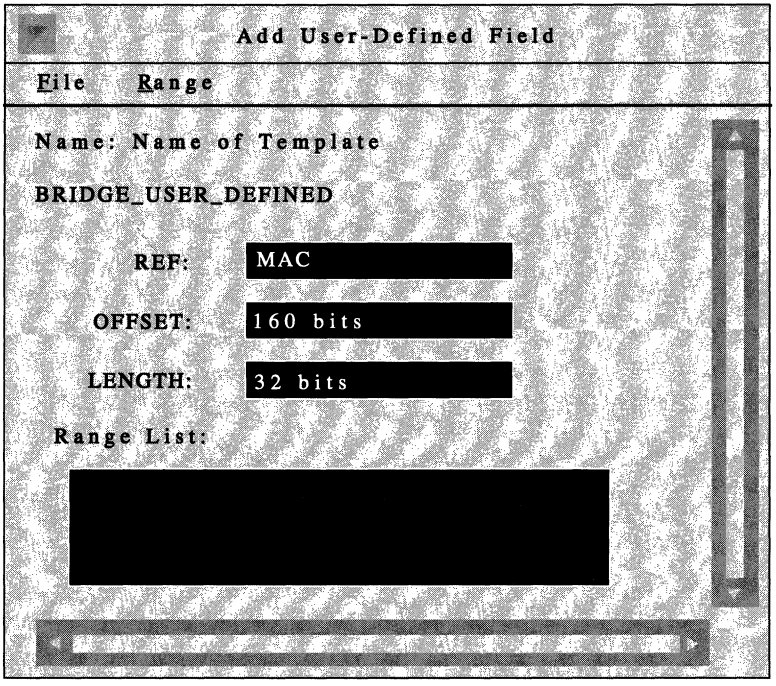


Figure 16-4. Add User-Defined Field Window

4. Specify the range to go with the field described by Reference, Offset, and Length.

   You specify the range the same way that you specify a range for a pre-defined filtering field; you simply select the Range/Add Range option. Then, you enter a minimum and maximum value in the appropriate boxes. In this case, you would specify 0x1234 for both the minimum and maximum values.

The procedures to add, delete, and edit ranges for a user-defined field are the same as the procedures for a pre-defined field. They are described in *Adding Ranges, Deleting Ranges*, and *Modifying Ranges*.

Using the Configuration Manager to Configure Filters

The following sections assume that you are familiar with protocol-specific filtering fields and actions, and with setting up a user-defined field if you intend to do so. They explain how to use the Configuration Manager to configure filters, which includes:

- Adding a filter to an interface, which may include any one or more of the following:
  - — Creating a template
  - — Copying a template
  - — Editing a template (its fields, ranges, and actions)
  - — Deleting a template
- Deleting a filter
- Editing a filter (its fields, ranges, and actions)

Adding a Filter to an Interface

This section describes adding a filter to an interface when there is no existing template that suits your needs (in this case, a template must be created). This process, described in the following steps, involves: naming the template, adding filtering fields and ranges to the template, adding actions to the template, then applying (saving) the template to the appropriate interface.

Start at the Wellfleet Configuration Manager Window, complete the following steps to first create a template, and then add a filter to the interface.

1. From the Protocols menu, select the appropriate protocol, and then the Interfaces option.

   The Protocol Interfaces List Window appears (see Figure 16-5). This window lists all the interfaces configured to run the protocol you chose in step 1. In this example, the Protocol Interfaces List Window for the Bridge is shown.

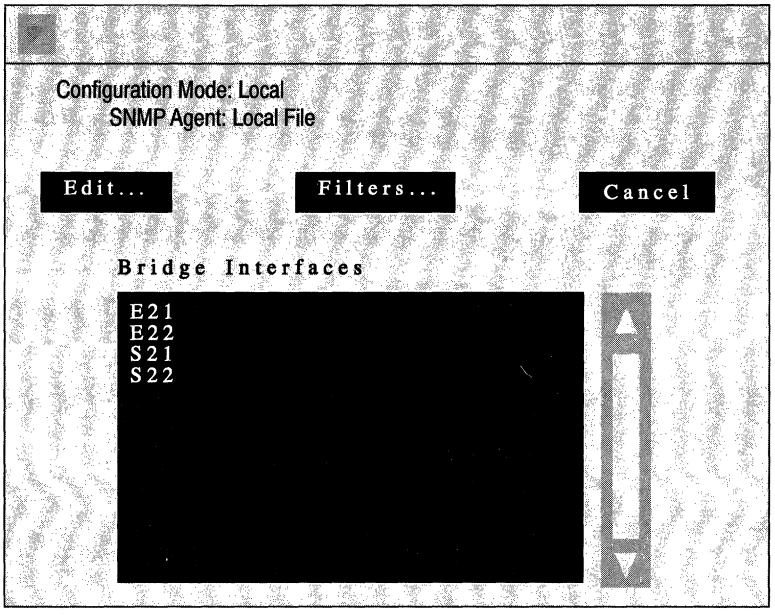


Figure 16-5. Protocol Interfaces List Window (Bridge)

2. Click the Filters button.

   The Protocol Interface Filters Window appears (see Figure 16-6). This window shows existing filters on the interfaces running this protocol. Assuming that this is the first template that you create, this window will not yet contain any filters.

3. Click the Add button.

   The Add Filter Window appears (see Figure 16-7).

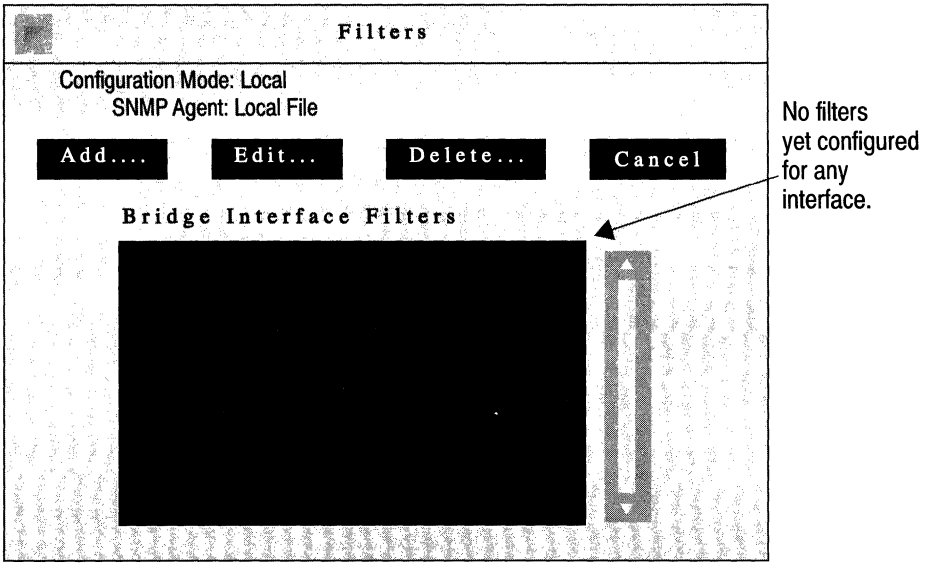


Figure 16-6. Protocol Interface Filters Window (Bridge)

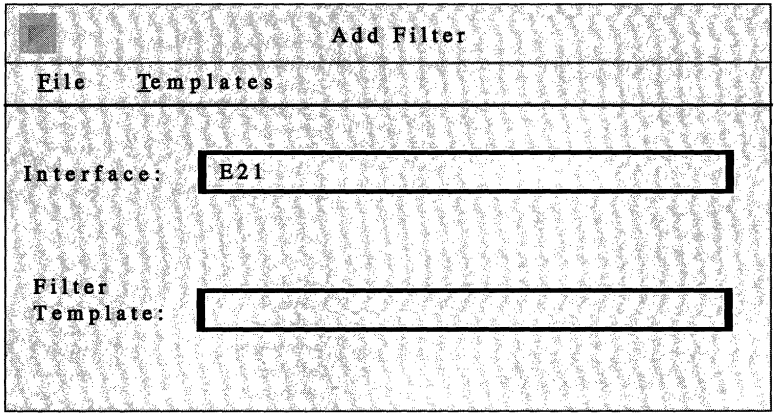


Figure 16-7. Add Filter Window

4. Select the Templates/Add Template option.

   The Edit Filter Window appears with all its fields blank (see Figure 16-8).

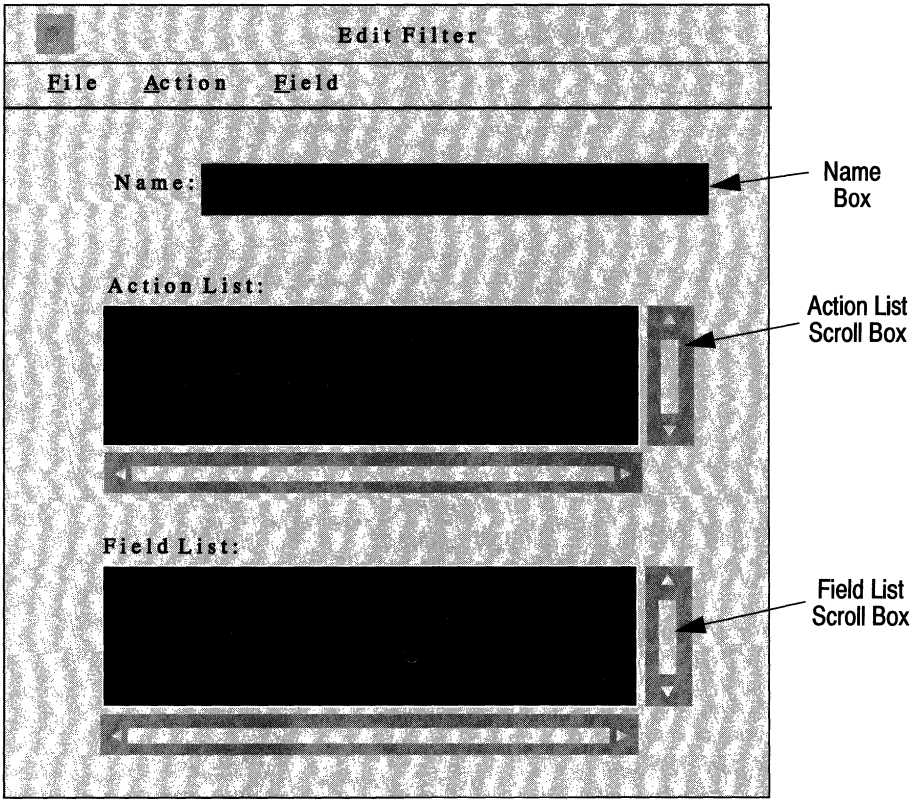


Figure 16-8. Edit Filter Window

5. Enter a name for this new template in the Name box.

   It is best to give descriptive names to your templates. For instance, in this example, the template will be called *Bridge01to03* because it will contain information for filtering bridge frames from certain MAC Source Addresses (0x0000A2000001 to 0x0000A2000003).

Next, you must add filtering fields and ranges to the template.

6. Select the Field/Add Field option, then select the protocol you chose in step 1(in this case, Bridge), which will be the only protocol highlighted.

   Another menu appears showing you the protocol-specific filtering field options (see Figure 16-9).

7. Select the field on which you wish to filter packets.

   In this example, the bridge protocol is first chosen. Then the MAC Source Address is chosen as a filtering field for this template.

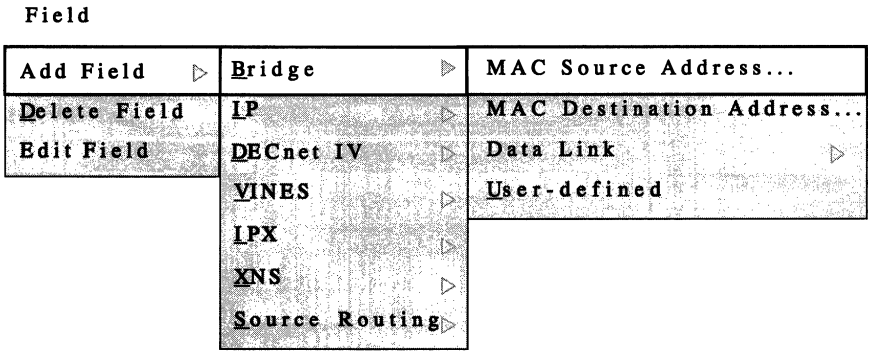


Figure 16-9. Choosing MAC Source Address as a Filtering Field

The Edit Field Window now appears (see Figure 16-10). For any field you choose, you need to specify at least one associated range.

8. Select the Range/Add Range option.

   The Edit Range Window appears (see Figure 16-11).

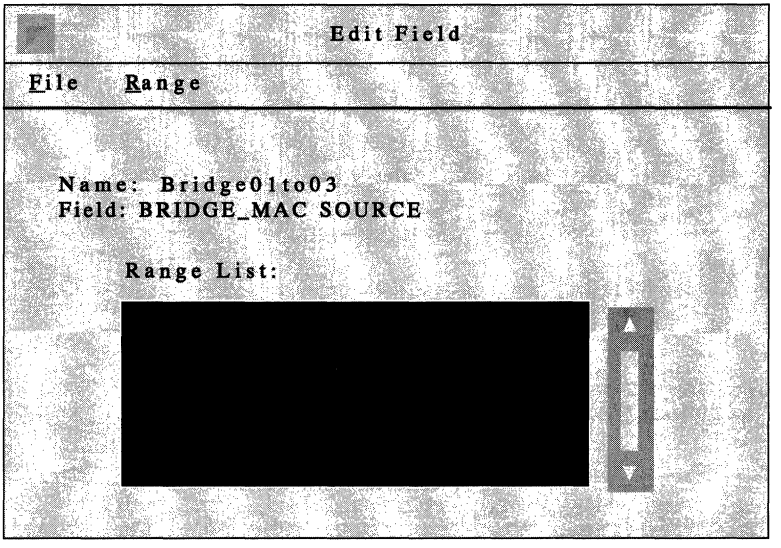


Figure 16-10. Edit Field Window

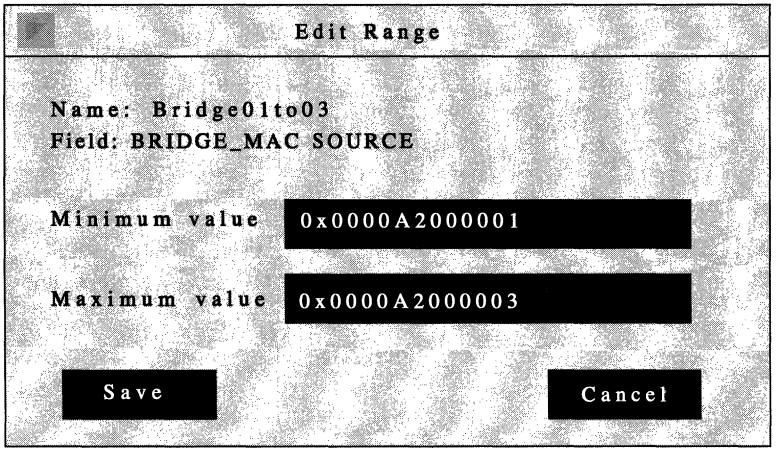


Figure 16-11. Edit Range Window

9. Specify the low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that value in both boxes. In this case, the MAC Source Address 0x0000A2000001 was specified as the minimum value, and the MAC Source Address 0x0000A2000003 was specified as the maximum value. Each incoming packet will be checked to see if its MAC Source Address falls into this range of addresses.

Note: When you enter values for minimum value and maximum value, the Configuration Manager assumes the value is a decimal number. If you want to enter a hexadecimal number, you *must* use the prefix 0x.

   The range you just specified will now appear in the Range List scroll box in the Edit Field Window. You can add up to 100 ranges per field by repeating steps 8 and 9.

10. When you are finished adding ranges to this field, select the File/Save option.

    You are returned to the Edit Filter Window.

    You can add additional filtering fields to this template; simply follow steps 6 through 10. Each field you add will appear in the Field List scroll box on the Edit Filter Window.

    Next, you must add one or more actions to your template.

11. From the Action menu, select the same protocol you selected in steps 1 and 6 (in this case, Bridge).

12. Select the Add Action option, then select the action you wish to impose on packets that match any of this template's filtering fields.

    In this example, Drop is chosen as the action for this template (see Figure 16-12). If you want to add another action (Log), repeat steps 11 and 12.

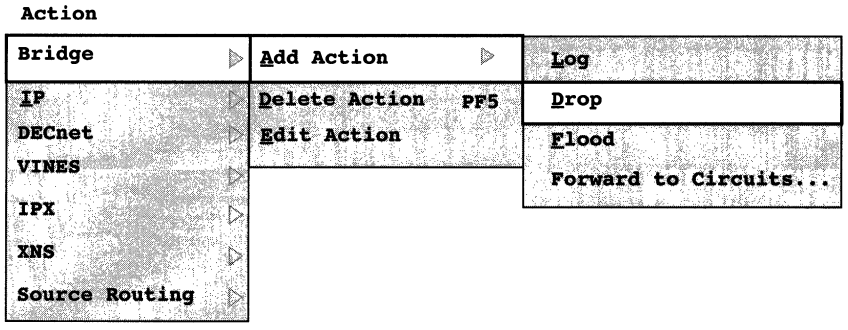


Figure 16-12. Choosing Drop as an Action

13. When you are finished adding actions to your template, select the File/Save option.

    This template is saved, and now appears in the Filter Template box in the Add Filter Window.

    Finally, you apply this template to an interface.

14. In the Add Filter Window, make sure the appropriate template appears in the Filter Template box; then, make sure the interface to which you want to apply it appears in the Interface box (see Figure 16-13).

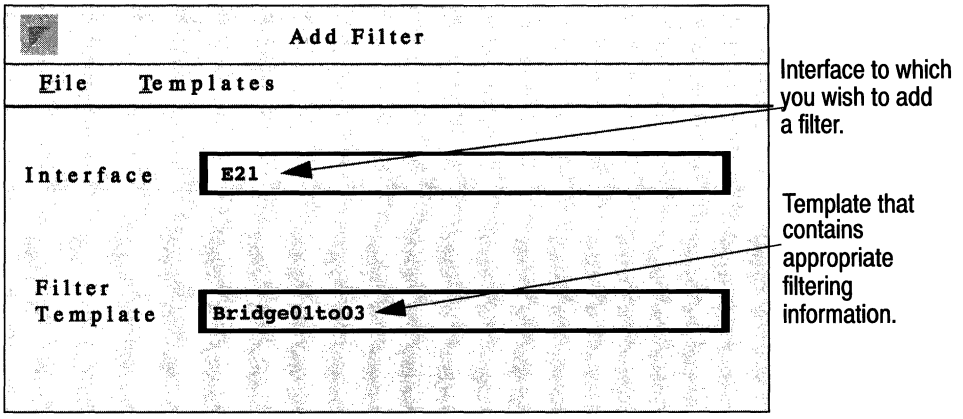


Figure 16-13. Specifying the Appropriate Template and Interface

15. When the appropriate template and interface are specified, select the File/Save option.

    In this example, the template called *Bridge01to03* is being applied to interface E21. The fields specified in this template now serve as a filter on interface E21. That is, each of E21's incoming packets will be checked to see if it matches any of *Bridge01to03*'s fields and associated ranges. If a match is found, the action (Drop, in this case) will be imposed on the packet; it will be discarded.

You are now returned to the Protocol Interface Filter Window. The filter you just created for interface E21 is listed in the Bridge Interface Filters scroll box. Assuming this is the first filter added to an interface, the window will appear as shown in Figure 16-14.

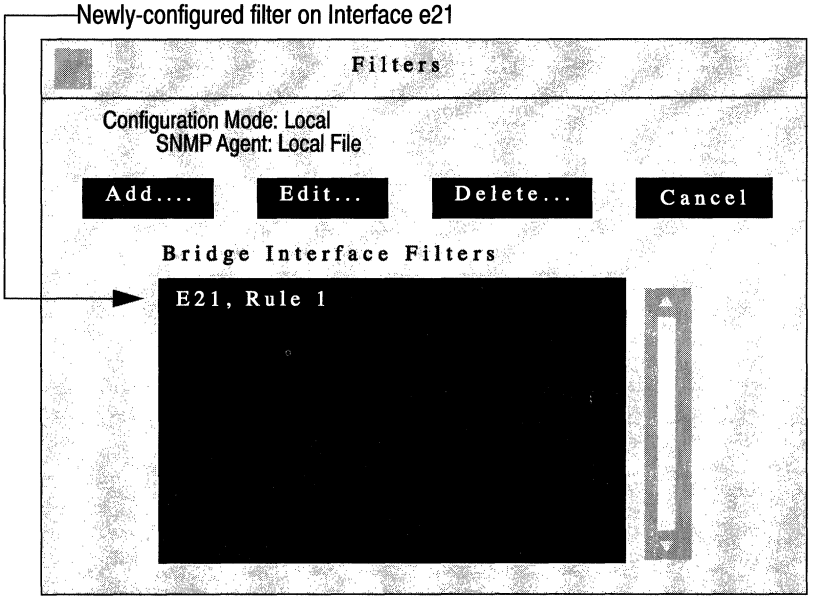


Figure 16-14. Newly Configured Filter in Bridge Interface Filters Window

Editing Templates

When you want to add a filter to an interface, you do not always have to create a new template. More often than not, you will be able to use existing templates to build new ones. For instance, suppose that on certain interfaces you want to filter all DECnet Phase IV frames with a Source Node between 19 and 23. Suppose also that there is a template named *DECSrc20* that contains fields and actions instructing all frames with a Source Node of 20 to be dropped. You can do one of two things:

- Specify the template (in this case *DECSrc20*), and edit it.

  Changes to this template will *not* affect interfaces to which it has already been applied.

- Copy the template, rename, and edit it.

  This creates an entirely new template (with the same fields, ranges and actions as those in the DECSrc20 template), that you can rename and edit to suit your needs.

These two options are discussed in the next two sections: *Copying a Template*, and *Specifying a Template.*

Note: You may also edit any template using a text editor. All templates are stored in a file called *template.flt*.

Copying a Template

Copying a template to a new name before you edit it is sometimes favorable, especially if you need to preserve the original template. To copy a template, complete the following steps. Then, proceed to the section entitled *Editing Fields, Ranges, and Actions* for instructions on editing this new template.

1. At the Wellfleet Configuration Manager Window, select the Protocols menu, then select a protocol, and finally, the Interfaces option.

   The Protocol Interfaces List Window appears.

2. Click the Filters button; the Protocol Interface Filters Window appears.

3. Click the Add button; the Add Filter Window appears (see Figure 16-15).

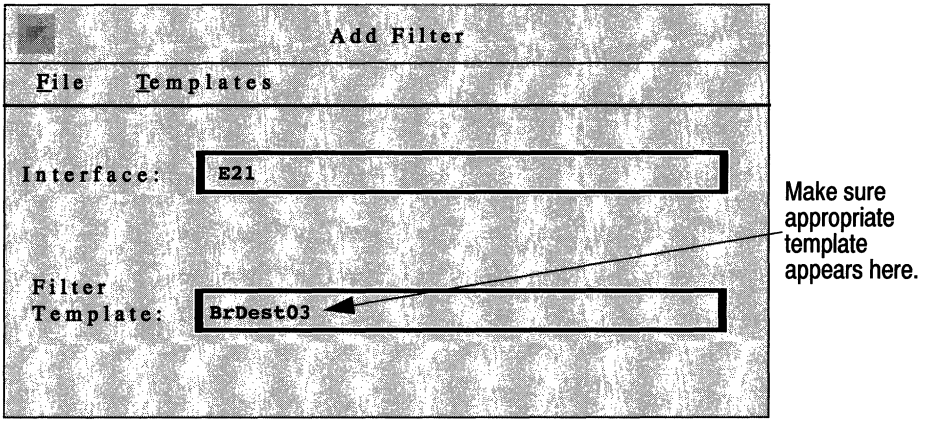


Figure 16-15. Add Filter Window

4. If the Filter Template box is displaying the name of the template you wish to copy, go to the next step. If the Filter Template is *not* currently displaying the name of the template you wish to copy, click on the box. A menu displaying all existing templates appears; choose the template you wish to copy.
5. Select the Templates/Copy Template option.

   The Copy Filter Template Window appears; see Figure 16-16.
6. Enter a name for the new template in the box provided.

   Remember that it is a good idea to give your template a name that reflects it contents.
7. Click the Save button.

   You are returned to the Add Filter Window. The name you just assigned to the new template appears in the Template Filter box.

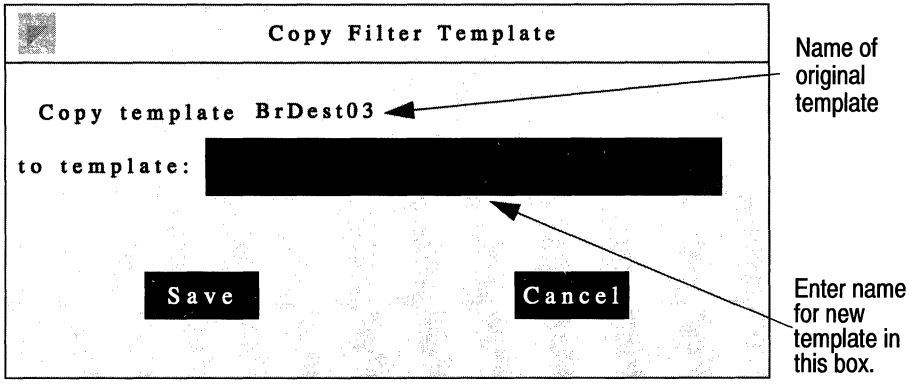


Figure 16-16. Copy Filter Template Window

8. Select the Templates/Edit Template option.

   The Edit Filter Window for the new template appears.

For instructions on editing this template, proceed to the section entitled *Editing Fields, Ranges and Actions*.

Specifying a Template

If you do not want or need to preserve the original template, you can simply edit it without first copying and renaming it. You simply need to specify it; start at the Wellfleet Configuration Manager Window, and complete the following steps:

1. From the Protocols menu, select a protocol, then select the Interfaces option.

   The Protocol Interfaces List Window appears.

2. Click the Filters button; the Protocol Interface Filters Window appears.

3. Click the Add button; the Add Filter Window appears (see Figure 16-17).

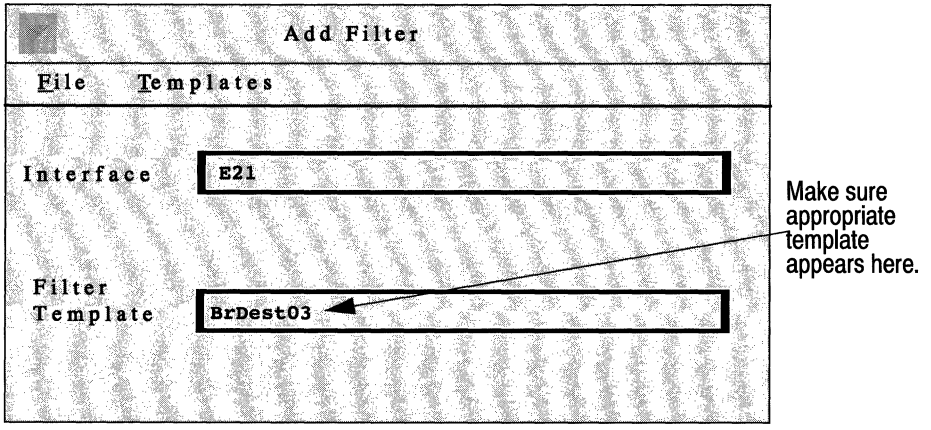


Figure 16-17. Add Filter Window.

4. If the Filter Template box displays the name of the template you wish to edit, go to the next step. If the Filter Template box does not display the name of the template you wish to edit, click on the box. A menu appears listing all existing templates; choose the appropriate one.
5. Select the Templates/Edit Template option.

   The Edit Filter Window for this template appears (see Figure 16-18). In this case, the template called *BrDest03* was chosen.

For instructions on editing this template, proceed to the next section, *Editing Fields, Ranges, and Actions*.

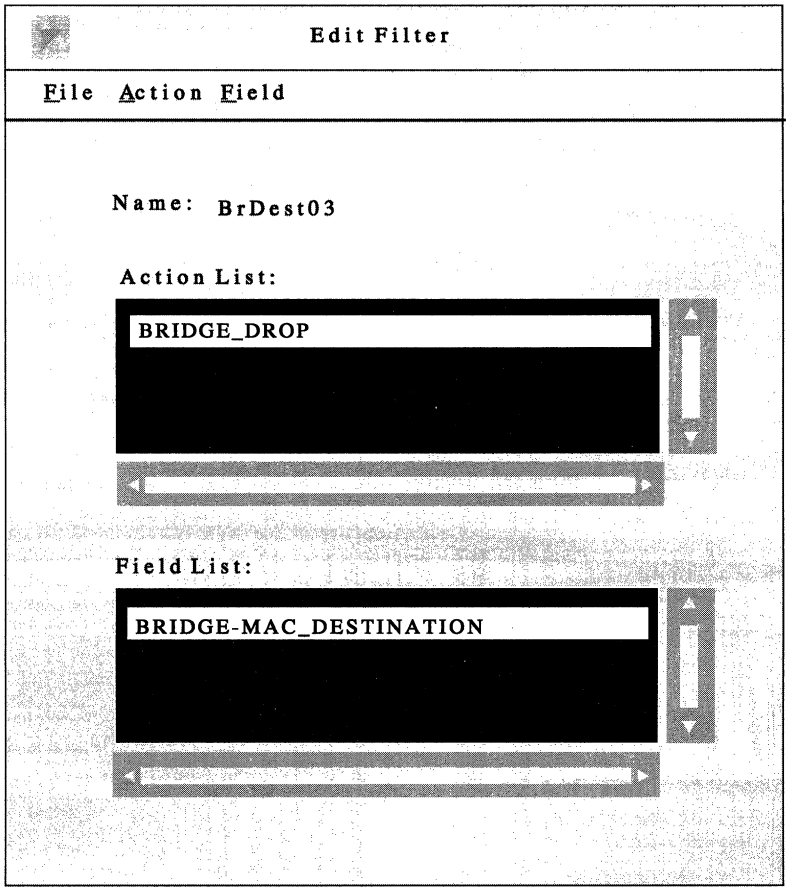


Figure 16-18. Edit Filter Window for a Specific Template

Editing Fields, Ranges, and Actions

Once you have either copied or specified a template, you can edit its fields, ranges, and actions. You have the following options, which are described in subsequent sections:

- Deleting or adding filtering fields
- Deleting, adding or modifying field ranges
- Deleting or adding actions

Deleting a Field

If you no longer want a field to be included in a template, follow these steps to remove it:

1. From the Field List scroll box, select the field you wish to delete.
2. Select the Field/Delete Field option.

   The Delete Field Window appears (see Figure 16-19). In this example, the MAC Destination Address field is being removed from the template.

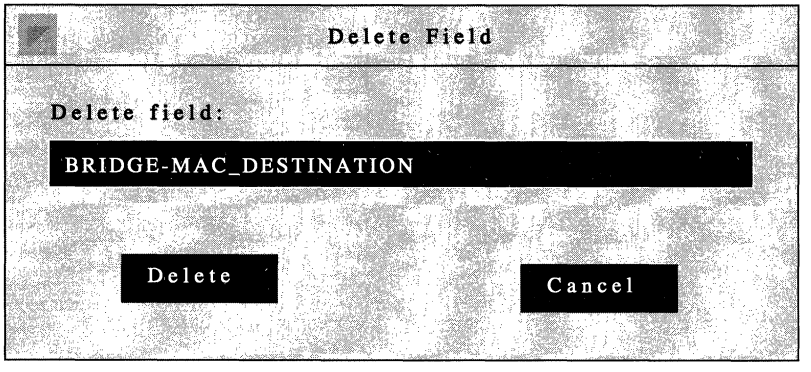


Figure 16-19. Delete Field Window

3. Click the Delete button.

You are returned to the Edit Filter Window. The field you just deleted no longer appears in the Field List scroll box.

Repeat this procedure for each field you wish to delete from a template.

Adding a Field

If you want to add a field to a template, complete the following steps. If you intend to add user-defined fields, refer to *Setting Up User-Defined Fields* (earlier in this chapter), which explains the special considerations of specifying user-defined fields.

1. Select the Field/Add Field option, then select the appropriate protocol (in this case, Bridge).

   Another menu appears showing you the protocol-specific filtering field options (see Figure 16-20).

2. Select the field on which you wish to filter packets.

   In this example, the MAC Source Address is chosen as the filtering field to be added to this template.

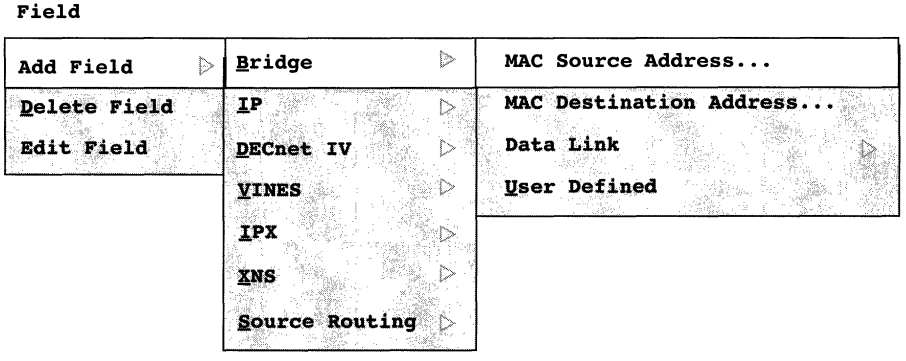


Figure 16-20. Choosing to Add MAC Source Address as a Filtering Field

The Edit Field Window appears (see Figure 16-21). For any field you choose, you must specify at least one range.

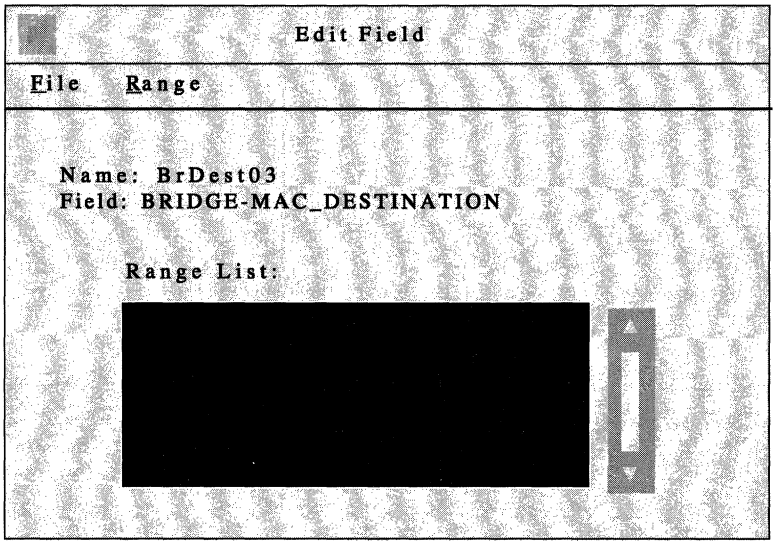


Figure 16-21. Edit Field Window

3. Select the Range/Add Range option.

   The Edit Range Window appears (see Figure 16-22).

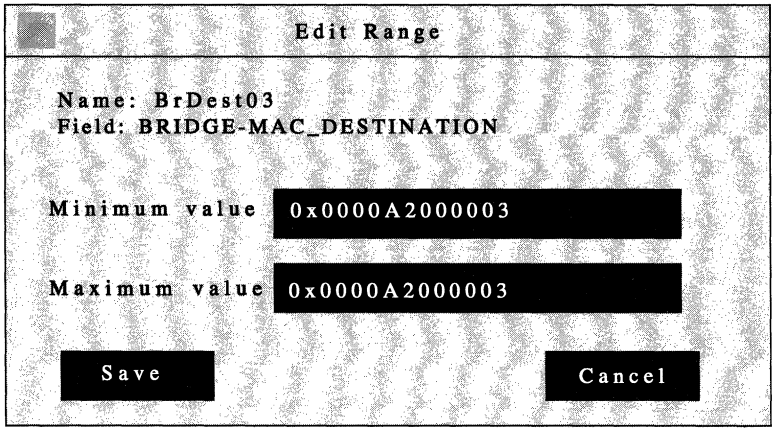


Figure 16-22. Edit Range Window

4. Specify the low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that same value in both boxes. In this case, the MAC Destination Address 0x0000A2000003 was specified as both the minimum and maximum value. Each incoming packet will be checked to see if its MAC Destination Address equals 0x0000A2000003.

Note: When you enter values for minimum value and maximum value, the Configuration Manager assumes the value is a decimal number. If you want to enter a hexadecimal number, you *must* use the prefix 0x.

   The range you just specified now appears in the Range List scroll box in the Edit Field Window. You can add up to 100 ranges per field by repeating steps 3 and 4 for each range you wish to add.

5. When you are finished adding ranges to this field, select the File/Save option.

   You are returned to the Edit Filter Window.

You can add filtering fields to this template. Simply follow steps 1 through 5 for each new field you wish to add. Each field you specify will appear in the Field List scroll box on the Edit Filter Window.

Deleting Ranges

If you need to delete a range from a template's field, complete the following steps.

1. From the Field List scroll box, select the field for which you wish to delete a range.
2. Select the Field/Edit Field option.

   The Edit Field Window appears (see Figure 16-23). It lists all of the ranges associated with this field.

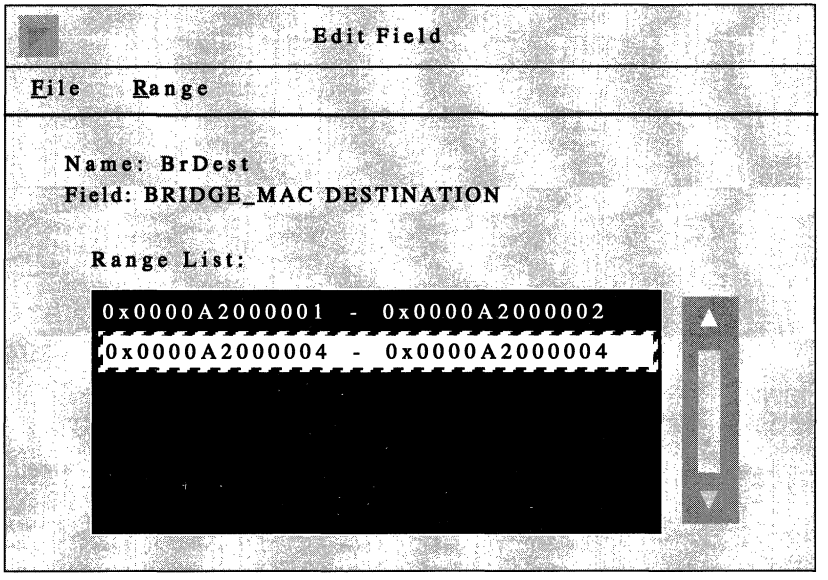


Figure 16-23. Edit Field Window

3. Select the range you wish to delete from this field.

   In this example, the range 0x0000A2000004 - 0x0000A2000004 is selected.

4. Select the Range/Delete Range option.

   The Delete Range Window appears (see Figure 16-24).

5. Click the Delete button.

   You are returned to the Edit Field Window. The range you just deleted from this field no longer appears in the Range List scroll box.

Repeat steps 1 through 5 for each range you wish to delete from a field.

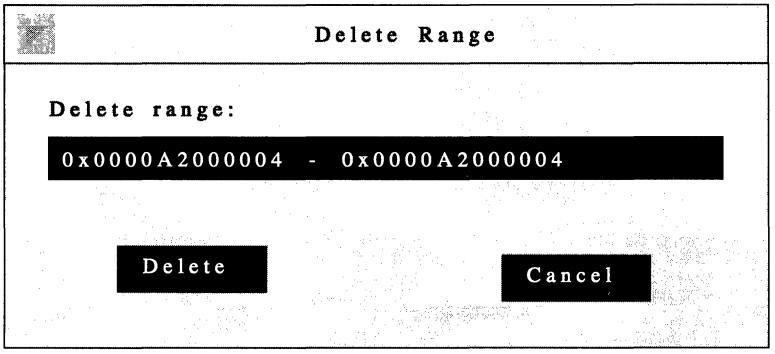


Figure 16-24. Delete Range Window

Adding Ranges

If you need to add a range to a template's field, complete the following steps.

1. From the Field List scroll box, select the field to which you wish to add a range.
2. Select the Field/Edit Field option.

   The Edit Field Window appears (see Figure 16-25). It lists all of the ranges associated with this field.
3. Select the Range/Add Range option.

   The Edit Range Window now appears (see Figure 16-26).

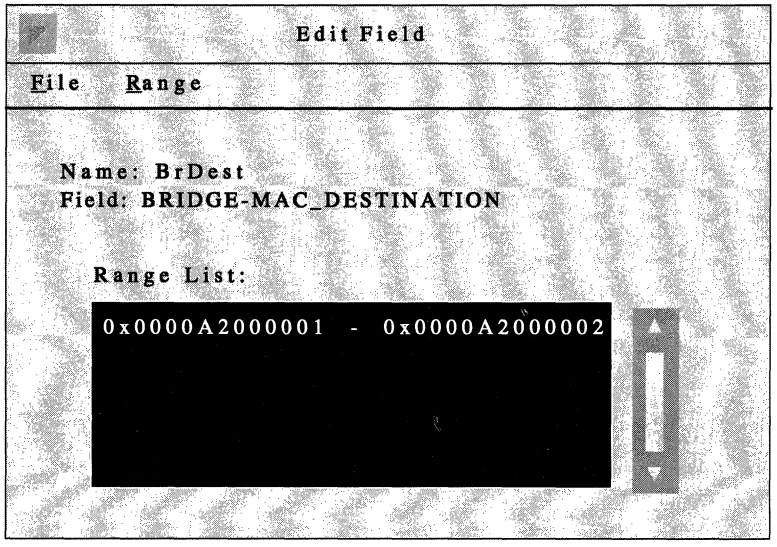


Figure 16-25. Edit Filter Window

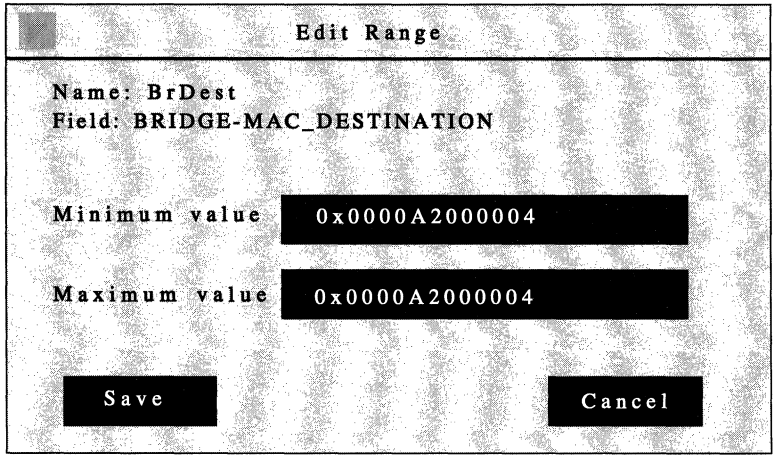


Figure 16-26. Edit Range Window

4. Specify the low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that same value in both boxes. In this case, the MAC Destination Address 0x0000A2000004 was specified as both the minimum and maximum value. Each incoming packet will be checked to see if its MAC Destination Address equals 0x0000A2000004.

Note: When you enter values for minimum value and maximum value, the Configuration Manager assumes the value is a decimal number. If you want to enter a hexadecimal number, you *must* use the prefix 0x.

   The range you just specified now appears in the Range List scroll box in the Edit Field Window. You can add up to 100 ranges per field by repeating steps 3 and 4 for each range you want to add.

5. When you are finished adding ranges to this field, select the Field/Save option.

   You are returned to the Edit Filter Window.

You can add additional ranges to any other field; simply follow steps 1 through 5. You can add up to 100 ranges per field.

Modifying Ranges

If you need to change a field's range, complete the following steps.

1. Select the appropriate field from the Field List scroll box.
2. Select the Field/Edit Field option.

   The Edit Field Window appears (see Figure 16-27). It lists all of the ranges that have been specified for this field.

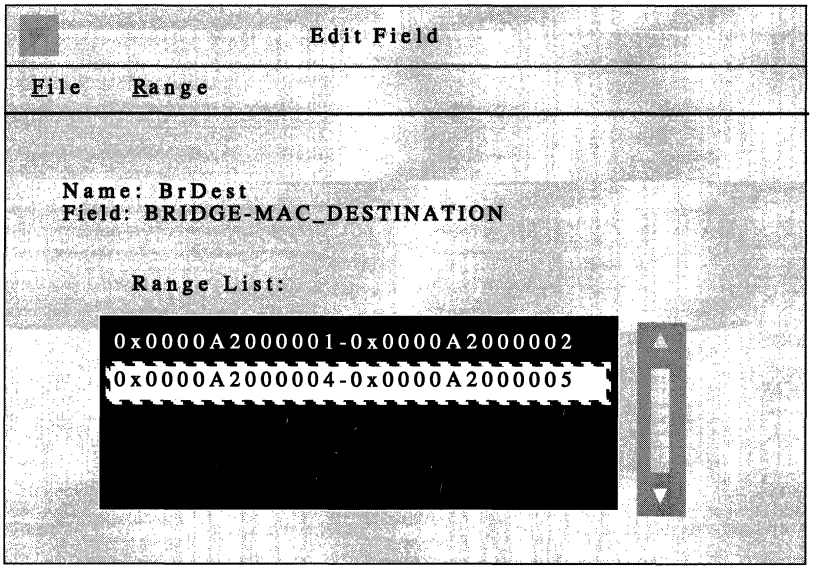


Figure 16-27. Edit Field Window

3. Select the range you wish to modify.

   In this example, range 0x0000A2000004-0x0000A2000005 is selected.

4. Select the Range/Edit Range option.

   The Edit Range Window appears (see Figure 16-28).

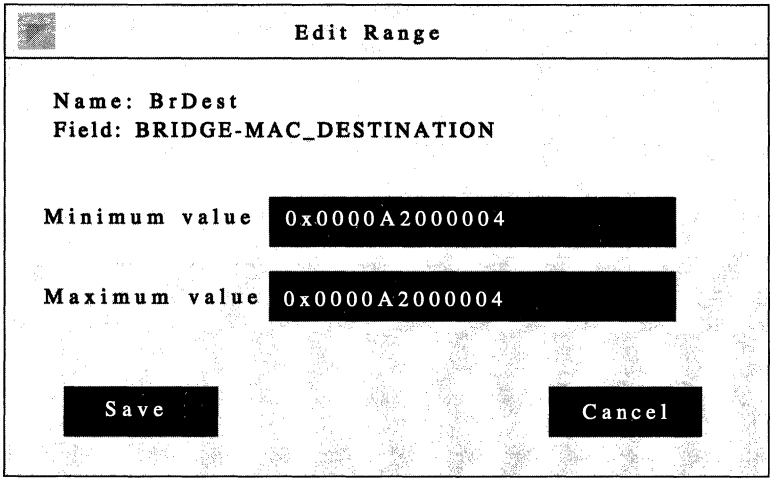


Figure 16-28. Edit Range Window

5. Specify the new low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that same value in both boxes. In this case, the MAC Destination Address 0x0000A2000004 was specified as both the minimum and maximum value. Each incoming packet will be checked to see if its MAC Destination Address equals 0x0000A2000004.

Note: When you enter values for minimum value and maximum value, the Configuration Manager assumes the value is a decimal number. If you want to enter a hexadecimal number, you *must* use the prefix 0x.

The range you just specified now appears in the Range List scroll box in the Edit Field Window. For each range you want to modify, repeat steps 3 through 5.

6. When you are finished modifying ranges for this field, select the Field/Save option.

   You are returned to the Edit Filter Window.

You can modify ranges for any other field; follow steps 1 through 6. You can add up to 100 ranges per field.

Deleting Actions

If you no longer want an action to be included in a template, follow these steps to remove it:

1. From the Action List scroll box, select the action you wish to delete.
2. From the Action menu, select the appropriate protocol (in this case, Bridge), then select the Delete Action option.

   The Delete Traffic Filter Action Window appears (see Figure 16-29). In this example, the Log action is being removed from the template.

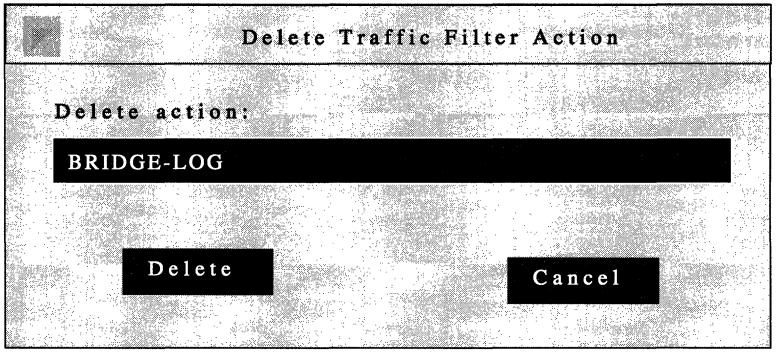


Figure 16-29. Delete Traffic Filter Action Window

3. Click the Delete button.

   You are returned to the Edit Filter Window. The action you have just deleted no longer appears in the Action List scroll box.

Repeat steps 1 through 3 for each action you wish to delete from a template.

Adding Actions

If you want to add an action to a template, follow these steps:

1. From the Action menu, select the appropriate protocol (in this case, Bridge).
2. Select the Add Action option.

   Another menu appears showing you the protocol-specific action options.
3. Select the action you wish to impose on packets that match any the template's filtering fields.

   In this example, Drop is chosen as the action to be added to this template (see Figure 16-30).

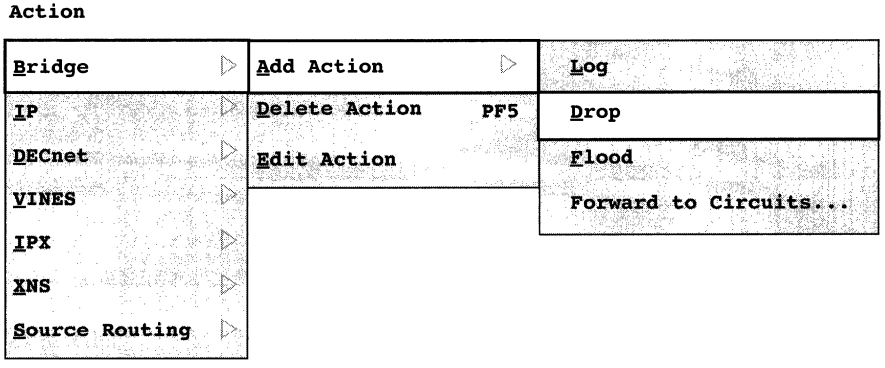


Figure 16-30. Choosing to Add the Drop Action to This Template

If you want to add another action, follow steps 1 through 3.

4. When you are finished adding actions to your template, select the File/Save option.

Modifying Actions

There are only three actions that you can modify; they are:

- Forward to Next Hop

  This action is an IP-specific action, and requires you to supply an IP address.

- Forward to Circuit List

  This action is a Bridge-specific action, and requires you to supply one or more MAC addresses.

- Direct IP Explorers

  This action is a Source Routing-specific action, and requires you to supply one or more IP addresses.

To modify an action, follow these steps:

1. From the Action List scroll box, select the action you want to modify.

   Only one of the following options will be available to you, depending on the protocol in which you are configuring filters: Forward to Next Hop, Forward to Circuit List, or Direct IP Explorers.

2. From the Action menu, select the appropriate protocol, either IP Bridge, or Source Routing.

3. Select the Edit option associated with the option you chose in step 1.

   The Modify Action Window appears showing the address(es) previously assigned to this action (see Figure 16-31). In this example, the Forward to Next Hop action is being modified.

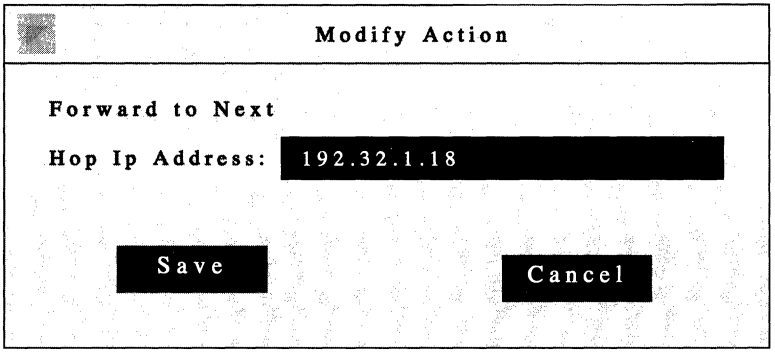


Figure 16-31. Modify Action Window for Forward to Next Hop

4. Enter the appropriate new IP address in the address box.
5. Click the Save button.

   You are returned to the Edit Filter Window. The modified action now appears in the Action List scroll box.

Deleting Templates

If you want to delete a template from your list of templates, begin at the Wellfleet Configuration Manager Window, and complete the following steps.

1. From the Protocols menu, choose any protocol, and then select the Interfaces option.

   The Protocol Interface List Window appears.
2. Click the Filters button.

   The Protocol Interface Filters Window appears.
3. Click the Add button.

   The Add Filter Window appears (see Figure 16-32).

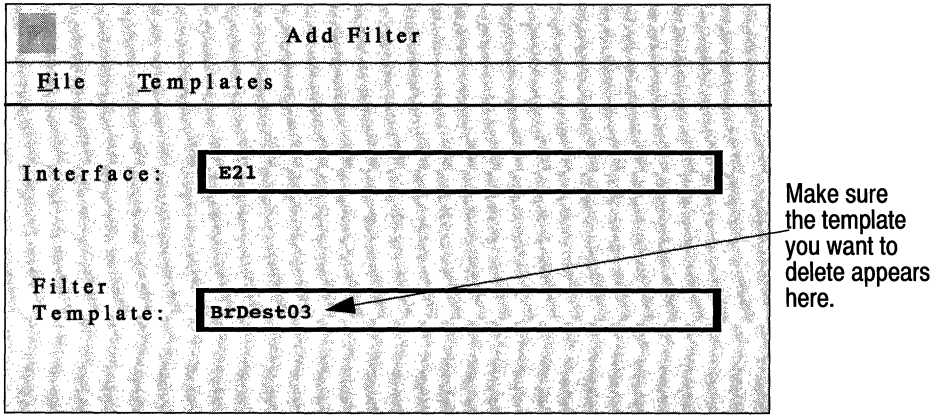


Figure 16-32. Add Filter Window

4. If the template that you wish to delete is displayed in the Filter Template box, go to the next step. If the template you wish to delete is not displayed in the Filter Template box, click on the box. A menu appears listing all existing templates; choose the template you wish to delete.
5. Select the Templates/Delete Template option.

   The Delete Filter Template Window appears (see Figure 16-33).

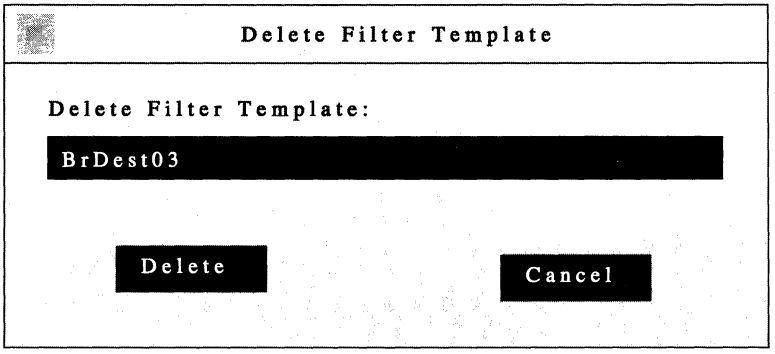


Figure 16-33. Delete Filter Template Window.

6. Click the Delete button.

   You are returned to the Add Filter Window. If you click on the Filter Template box, the filter you just deleted is no longer included in the list of templates.

Deleting a Filter

If you want, you can delete filters from individual interfaces. When you do, it affects only the interface from which the filter is removed. To delete a filter from an interface, complete the following steps:

1. Start at the Wellfleet Configuration Manager Window, and select the Protocols menu.
2. Select the appropriate protocol, then click the Interfaces button.

   The Protocol Interfaces List Window appears (see Figure 16-34). This window lists all the interfaces configured to run the protocol you chose in step 1. In this example, the Protocol Interfaces List Window for the Bridge is shown.

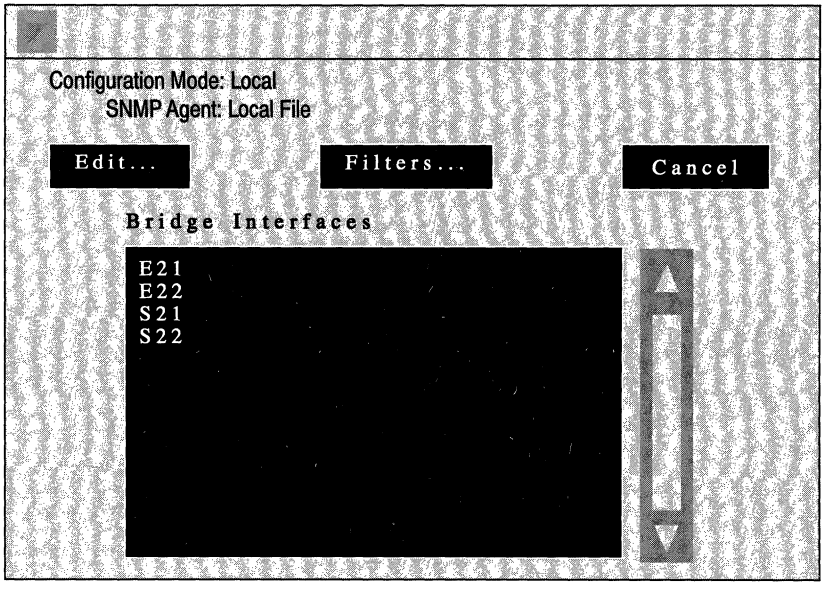


Figure 16-34. Protocol Interfaces List Window (Bridge)

3. Click the Filters button.

   The Protocol Interface Filters Window appears (see Figure 16-35). This window shows filters existing on the interfaces running this protocol.

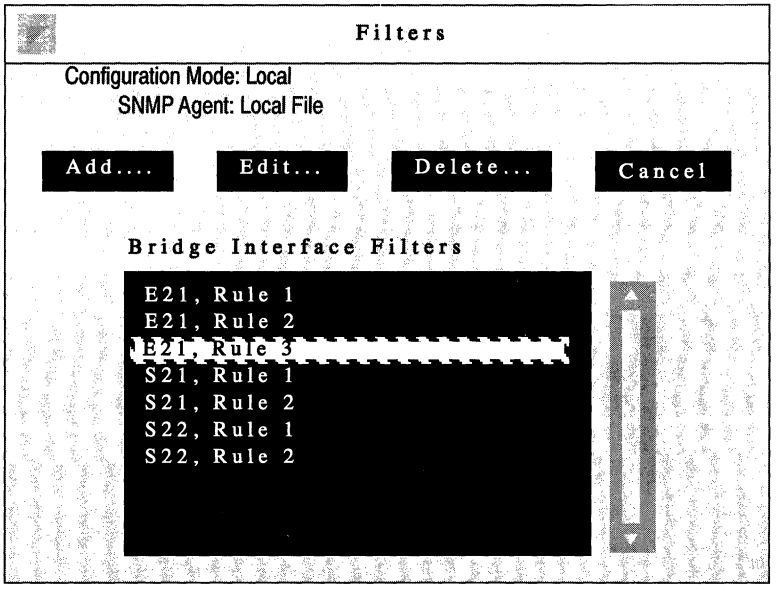


Figure 16-35. Protocol Interface Filters Window (Bridge)

4. From the Interface Filters scroll box, select the interface/filter that you wish to delete.

   For example, in this case filter Rule 3 is being deleted from interface E21.

5. Select the Delete button.

   The Delete Filter Window appears (see Figure 16-36).

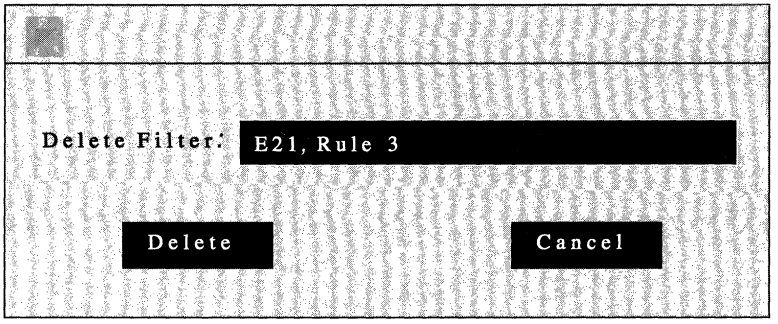


Figure 16-36. Delete Filter Window

6. Click the Delete button.

   The filter is deleted from the interface and no longer appears in the Interface Filters scroll box on the Protocol Interface Filters Window.

Editing a Filter

If you want, you can edit filters on individual interfaces. When you do, only the filter on that specific interface is affected. To edit a filter, start at the Wellfleet Configuration Manager Window and follow these steps.

1. From the Protocols menu, select the appropriate protocol, then select the Interfaces button.

   The Protocol Interfaces List Window appears (see Figure 16-37). This window lists all the interfaces configured to run the protocol you chose in step 1. In this example, the Protocol Interfaces List Window for the Bridge is shown.

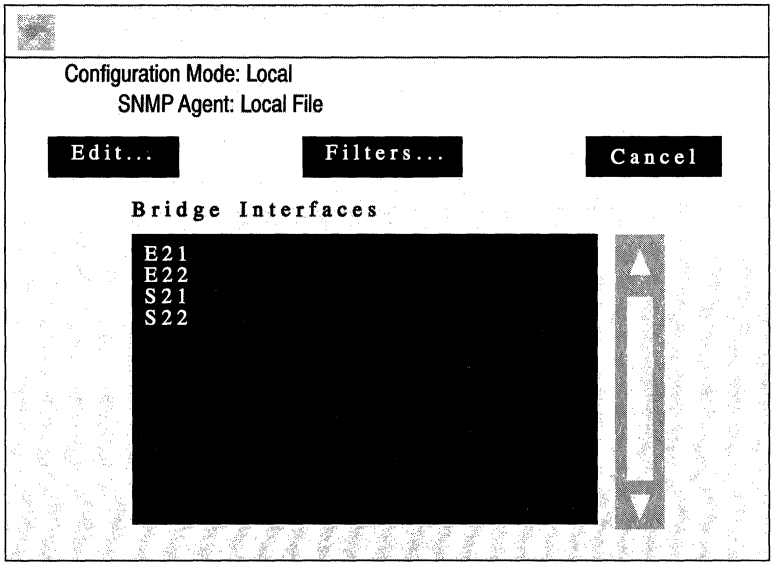


Figure 16-37. Protocol Interfaces List Window (for the Bridge)

2. Click the Filters button.

   The Protocol Interface Filters Window appears (see Figure 16-38). This window shows filters existing on the interfaces running this protocol.

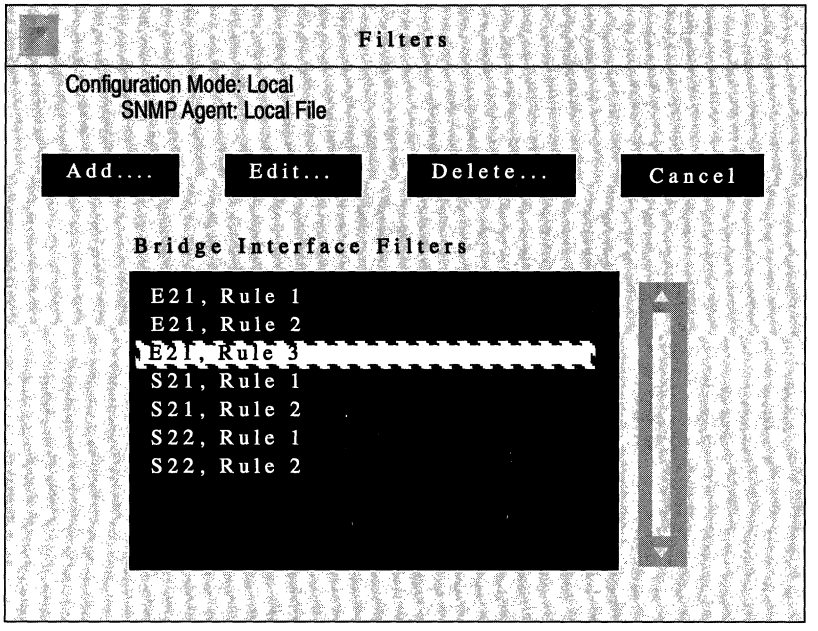


Figure 16-38. Protocol Interface Filters Window (Bridge)

3. From the Interface Filters scroll box, select the interface/filter that you wish to edit.

   For example, in this case filter Rule 3 on interface E21 is selected.

4. Click the Edit button.

   The Edit Filter Window for this interface/filter appears. Editing a filter's fields, ranges and actions is the same as editing a template's fields, ranges, and actions. However, when you save your changes, it will affect the filter on this interface only. Refer to the section entitled *Editing Fields, Ranges, and Actions* for instructions on how to edit this filter.

Chapter 17

Configuring Protocol Prioritization

List of Figures

List of Tables

Configuring Protocol Prioritization

About this Chapter

This chapter provides the following:

- An overview of protocol prioritization
- A description of data link header and IP header fields, field ranges, and actions
- An explanation of how to use the Configuration Manager to configure protocol prioritization, which includes:
  - Adding a content-based priority filter to an interface
  - Deleting a content-based filter from an interface
  - Editing a content-based priority filter
  - Editing length-based priority filters
  - Editing protocol prioritization interface-specific parameters

You should read this chapter if you are responsible for configuring protocol prioritization for your network.

What is Protocol Prioritization

Protocol prioritization is a mechanism that allows you to assign priorities to the type of traffic transmitted on an individual synchronous line interface. It enables the router to transmit certain packets, based on their assigned priority, before transmitting other packets. Priority is assigned based on packet type and any other field that can be identified by an offset in the packet. Also, a packet can be prioritized by its length.

Depending on whether or not you assigned priority to a certain type of packet, a packet will be held in one of three priority queues: high, normal, or low. Figure 17-1 illustrates a variety of packets, that have converged at a synchronous line interface, being sorted by priority into the appropriate priority queues. Note that the packets with no assigned priority automatically go into the normal priority queue.

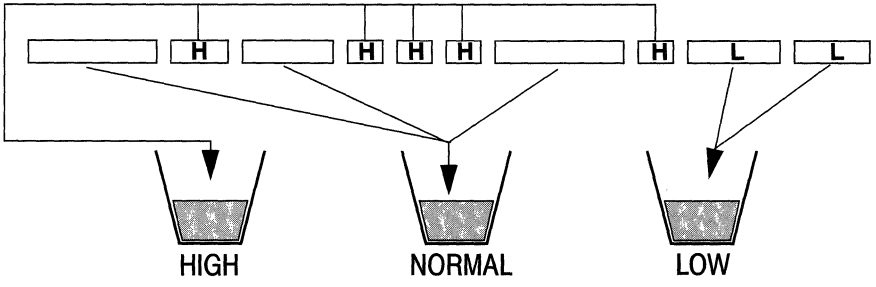


Figure 17-1. Traffic Being Sorted By Priority into the Appropriate Priority Queue

Generally, when traffic is transmitted, traffic in the high priority queue takes precedence over traffic in the normal priority queue, and traffic

in the normal priority queue takes precedence over traffic in the low priority queue.

Two other configurable values in the protocol prioritization scheme affect the transmission of traffic; they are queue depth and latency (which is actually the line delay). Queue depth dictates the number of packets a priority queue can hold. The latency value dictates the maximum time delay that high priority traffic can experience. A subsequent section, *Tuning Protocol Prioritization For Your Network*, provides a more in-depth description of these values and how they are useful in tuning protocol prioritization for your network.

Why Would You Use Protocol Prioritization

Protocol prioritization is useful when the synchronous line resource is being shared with different kinds of traffic. In this environment, time-sensitive, smaller packet traffic (for example, DEC LAT, or IP Telnet), usually gets delayed during the transmission of larger packet traffic, such as file transfers. This delay results in loss of connections, and poor terminal response (slow echoing of keystrokes and slow response to commands).

Protocol prioritization solves these problems by allowing you to assign a high priority for the time-sensitive protocol traffic that is being affected, thus allowing this type of traffic to be transmitted before normal and low priority traffic.

Another use for protocol prioritization is to expedite transmission of traffic coming from a particular source, or going to a certain destination. For example, if you wanted all traffic from the workstation with the Source MAC address 00:00:A2:00:00:12 to take precedence over other traffic, you could assign a high priority to any traffic with a source address that matches 00:00:A2:00:00:12.

As an example of how protocol prioritization is useful, consider the network shown in figure 17-2, and suppose that the following traffic conditions are typical:

- File transfers from VAX1 to VAX 2
- File transfers from SUN1 to SUN2
- LAT sessions from TS1 to VAX2
- Telnet sessions from SUN3 to SUN4

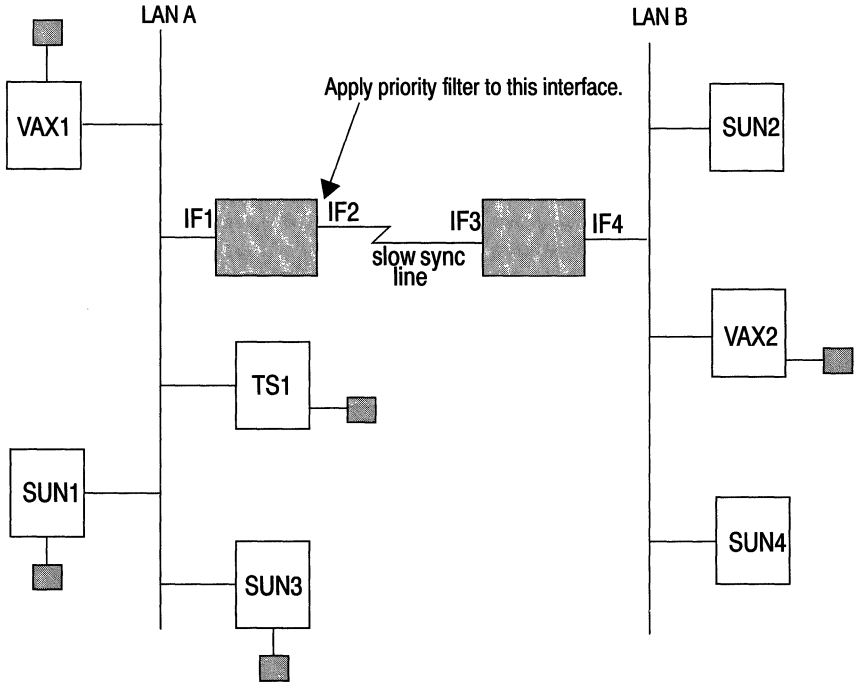


Figure 17-2. Sample Network

You would need to set up two priority filters to ensure that the LAT and Telnet traffic from LAN A are expedited to LAN B, not delayed by the file transfers going from LAN A to LAN B. These filters would need to be applied to Interface 2, as prioritization is concerned with outbound traffic, and the direction of the traffic flow is from LAN A to LAN B. The following priority filters would be necessary. Note that these are content-based priority filters that use pre-defined fields. The difference between content-based and length-based priority filters is discussed in *Priority Filters*.

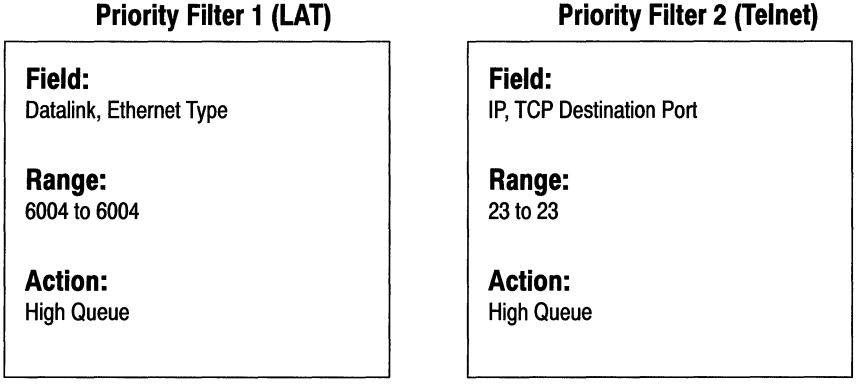


Figure 17-3. Priority Filters Allotting High Priority Status to LAT and Telnet Traffic

Tuning Protocol Prioritization For Your Network

As mentioned previously, there are two values that you use to tune protocol prioritization for your network. They are the queue depth and the latency value. The subsequent sections explain these values and how to use them to your advantage.

Queue Depth

Queue depth indicates the capacity of the priority queues; it is configured in terms of the number of packets that the queue can hold. The default value is 20 packets. This value is set regardless of packet size.

When you configure the queue depth, you are actually assigning buffers (which hold the packets), to the queue. Sometimes you may find that you have allocated too many, or not enough buffers to a priority queue. This can be determined by examining protocol prioritization statistics. There are two counters that pertain to the queue depth and that allow you to determine whether the queue depths are appropriate for your typical traffic flow. They are:

- The HiWater Packets Mark

  The HiWater Packets Mark reflects the fullest that the priority queue has gotten at any one time. There is a HiWater Packets Mark counter for each priority queue.

- The Clipped Packets Count

  The Clipped Packets Count records how many packets have been discarded for a particular priority queue. Packets that are destined for a full priority queue (the packets being held in queue equals the queue depth) are discarded. There is a Clipped Packets Count counter for each priority queue.

Generally, if a priority queue's Clipped Packets Count is high, and its HiWater Packets Mark is close to or equal to its queue depth, there are not enough buffers assigned to that queue. To illustrate the usefulness of these statistics, consider the following example.

Suppose that you kept the default queue depth (20 packets) for all of your priority queues. Upon inspection of the statistics, you see that the high priority queue's Clipped packets Count equals 226, and its HiWater Packets Mark equals 20. See Figure 17-4. This means that your high priority queue has been full at least once (probably more than once), and that 226 packets have been discarded. Conclusion: there are not enough buffers assigned to the high priority queue for the amount of high priority traffic on this interface.

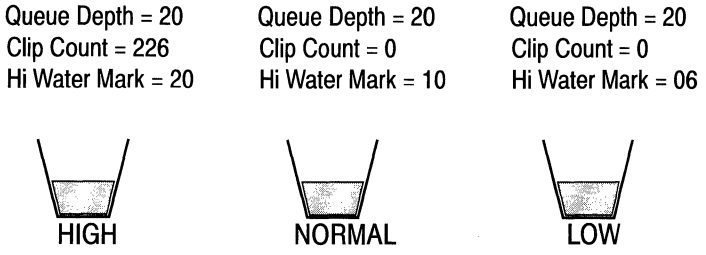


Figure 17-4. Clipped Packets Counts and HiWater Packets Marks for the Priority Queues

You can do one of two things to alleviate this problem. The first option is to reconfigure the queue depths. Looking at the statistics of the normal and low priority queues, you find that the low priority queue has a Clipped Packet Count equal to 0, and a HiWater Packets Mark equal to 6. This means that there have never been more than 6 packets at any one time in the low priority queue and therefore no packets have been discarded.

At this point, you may choose to reconfigure the low priority queue depth to 10, and increase the high priority queue depth to 30. In order to see if this reallocation solves the problem, you reset the Clipped Packets Count and HiWater Packets Mark counters (this is done on the Site Manager's Protocol Prioritization Statistics Window when you select the Zero Totals option), then check them again later.

Your second option, the one you may have to resort to should the first option not remedy the problem, is to remove the high priority status of some of the traffic types that are configured as high priority traffic. A network manager should be selective in assigning high priority status. If there are too many traffic types with high priority status, the high priority traffic could starve the normal and low priority traffic.

Latency

Protocol prioritization guarantees you a configurable delay for your high priority traffic. You configure this line delay through the latency value. Latency is associated with the transmit queue (the queue that scans and drains the priority queues and transmits the traffic). It dictates how many normal or low priority bytes can be on the transmit queue at any one time, and therefore, the greatest delay that a high priority packet can experience.

Latency is based on the line speed of the attached media. For a given line speed, the number of bits that can be queued to the transmit queue at any one time is determined by the configured latency value.

The latency, or line delay, is best expressed by the following formula:

$$\text{Latency} = \frac{\text{Bits queued}}{\text{Line speed (bits/sec)}}$$

The default value for Latency is 250 ms. This value allows good throughput and also preserves rapid terminal response (rapid echoing of keystrokes and timely response to commands), over most media. See Table 17-1, which shows the number of packets of a given size that can be queued to the transmit queue in order to achieve a latency of 250 ms over different types of media. Note that the information in this table is based on 90% bandwidth utilization.

Table 17-1. Maximum Number of Packets Queued to Achieve 250 ms Latency

Number of Packets Queued Latency = 250 ms	T1 1.544 Mb/s	56 Kb/s	9.6 Kb/s
60 bytes (small pkt)	643pkts	23 pkts	4 pkts
1514 bytes (Ethernet)	25 pkts	0 (.92) pkts	0(.16) pkts
4096 bytes (FDDI)	9 pkts	0 (.34) pkts	0(.06) pkts

Note: The latency value is user configurable; however, keep in mind that if you configure a higher latency value (thus allowing more room on the transmit queue), the throughput becomes greater, but you sacrifice the crispness of terminal response. Wellfleet recommends accepting the default value of 250 ms.

How Protocol Prioritization Works

As the Wellfleet router operates, network traffic from a variety of sources converges at the synchronous line interface. This traffic is placed into the high, normal or low queue according to the priority filter(s) that you have configured on this interface. Or, the traffic is clipped. Protocol prioritization then uses a dequeuing algorithm to govern the draining of the priority queues and the transmission of traffic. Figure 17-5 illustrates the relationship between the priority queues and the transmit queue.

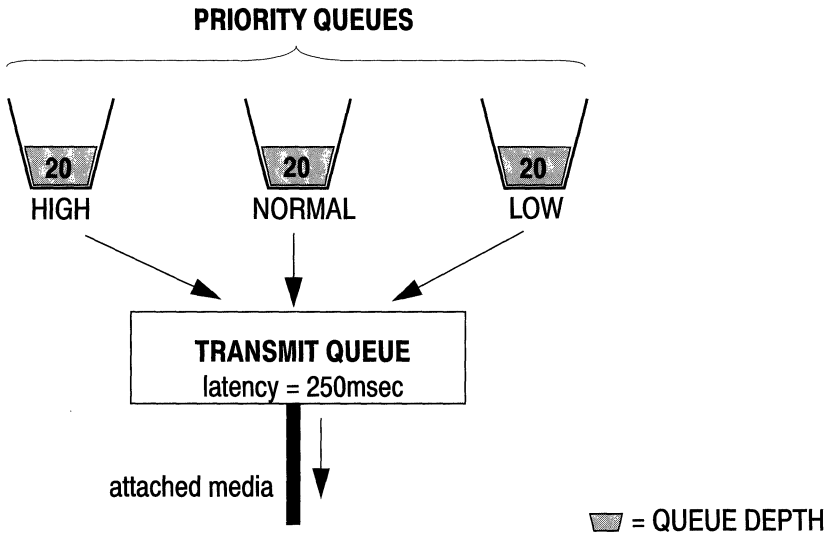


Figure 17-5. Relationship Between Priority Queues and Transmit Queue

The Dequeuing Algorithm

The protocol prioritization dequeuing algorithm works in the following way: First, the transmit queue scans the high priority queue. If there is traffic in the high priority queue, all packets up to the hardware limit, if there are that many packets, are emptied into the transmit queue and transmitted.

The hardware limit is the maximum number of packets that can be queued to the transmit queue at one time. This is a not a configurable number.

At this point, there are two considerations:

- Was the hardware limit reached?
- Was latency reached?

If the answer to either one of these questions is *yes*, then the transmit queue scans and empties traffic from the high priority queue again. If the answer to both of these questions is *no*, then the transmit queue scans the normal priority queue.

One of two conditions will exist in the normal priority queue:

- If there is traffic in the normal priority queue, a certain number of bytes, up to the latency value, are emptied into the transmit queue and transmitted.

  If the latency is reached, then the high priority queue is again scanned.

  If latency is not reached at this time (there were not enough bytes in the normal priority queue to fill the transmit queue), then the low priority queue is scanned.

- If there is no traffic in the normal priority queue, the low priority queue is then scanned.

Again, one of two conditions will exist in the low priority queue:

- If there is no traffic in the low priority queue, then the transmit queue starts at the beginning, scanning the high priority queue.

- If there is traffic waiting in the low priority queue, a certain number of bytes (up to the latency value, if there are that many bytes), are emptied into the transmit queue and transmitted.

  At this point, whether or not the latency value is reached, the transmit queue starts at the beginning, scanning the high priority queue.

Figure 17-6 illustrates the dequeuing algorithm.

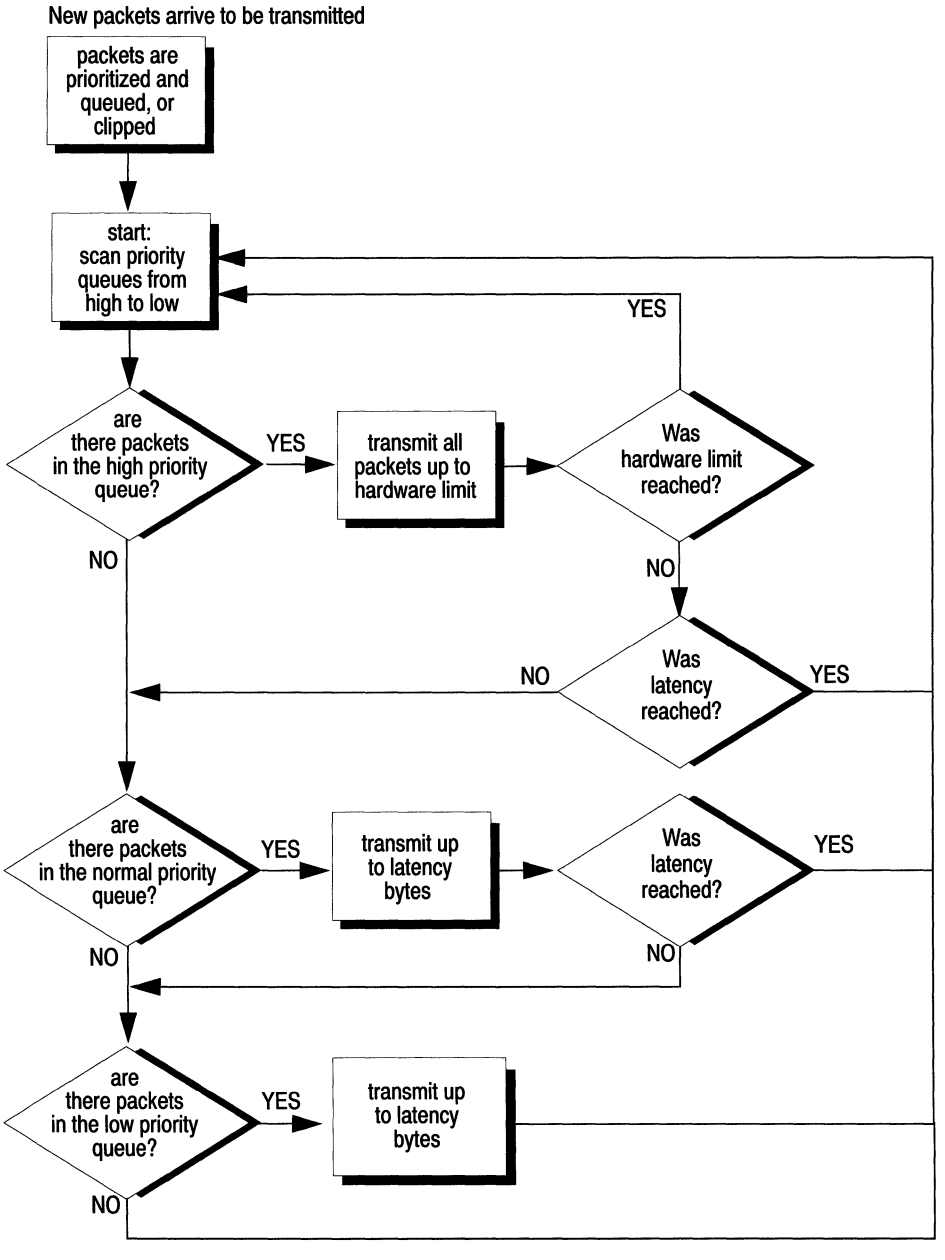


Figure 17-6. The Dequeuing Algorithm

Priority Filters

When you configure protocol prioritization, you are configuring a priority filter. A priority filter is simply a set of conditions and an action that you apply to a circuit or interface. Protocol prioritization allows you to configure two different types of priority filters: length-based and content-based. One is configured differently than the other.

- *A Length-based priority filter* assigns priority for a particular type of traffic (that you specify), based on packet length. Length-based priority filters are created only at the circuit level. That is, when you first configure a circuit (name it, add protocols to it), you must also set up any length-based priority filters you want for that circuit. Length-based filters cannot be added to or deleted from an interface at the protocol level; however, they can be edited at this level.

  Creating and deleting a length-based filter is discussed in *Configuring Circuits*. Editing a length-based filter is discussed later in this chapter.

- *A Content-based priority filter* assigns priority through either the datalink or IP header, and is based on either pre-defined or user-defined fields. Content-based filters can be configured either at the circuit level or the protocol level. This type of priority filter allows you to create templates (files that contain priority filter information). This means that you do not have to recreate the same priority filter. You simply create it once, name and save it. Then, you can apply it to any interface that you choose.

  Creating, editing and deleting a content-based filter are discussed later in this chapter.

Templates and Filters

It is important for you to understand the difference between a template and a filter. A template is simply a file that holds specific filtering information (fields, ranges, and actions). A filter is created from a template when you apply (save) the template to an interface.

You can apply a single template to as many interfaces as you want; see Figure 17-7. Once a template is created, it exists for future use unless you delete it.

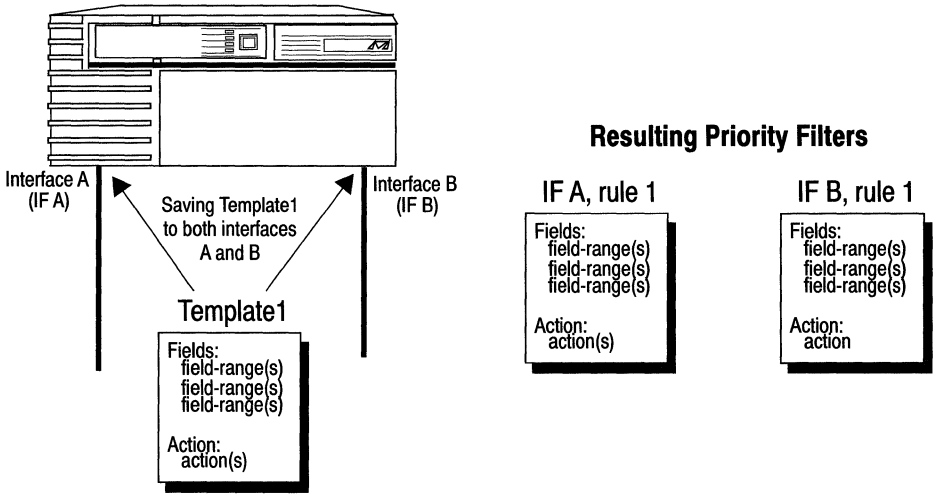


Figure 17-7. Using a Template to Create Priority Filters

Generally, when you create a template, you first assign it a one-word name. It is a good idea to give your template a descriptive name. For example, if you are building a template that is going to contain filtering information instructing the interface to queue all LAT traffic to the high queue, then you may want to name the template something like *LAThigh*.

Once you have named your template, you select the fields and assign ranges for which each packet will be checked. You then select the action(s) that will be imposed on any packet that matches at least one range for every field in the filter. After you have specified this information, you save it, thus creating a template. When you add this template to an interface, you have created a content-based priority filter on that interface.

When you want to add a content-based priority filter to an interface, you have several options:

- If there is an existing priority filter on the interface that contains filtering instructions similar to what you want, you may edit this filter and save it (see *Editing a Content-Based Priority Filter*). The changes to the filter are valid only on this interface.
- If there is a template that contains the exact filtering instructions that you desire for this interface, you can apply (save) that template to this interface.
- If there is a template that contains filtering instructions similar to what you want, you can copy the template, rename, and edit it (see section entitled *Editing a Template*). When you save the changes, you have created a new template. You can now apply this template to any interface for which its filtering instructions are appropriate.
- If there is no template containing filtering instructions similar to what you want for this interface, you must create a template from scratch.

It is this last case that is discussed in *Adding a Content-Based Priority Filter to an Interface*.

Filtering Fields, Ranges and Actions

As previously described, all filters are created from templates (files that hold filtering information), which consist of three components:

- Field

  A filtering field is a part of a packet, frame or datagram header that you specify to be checked on each incoming frame. Each filtering field has one or more ranges associated with it.

- Range

  A range is associated with a filtering field. There must be at least one range per field. A range can consist of just one value, or it can be a set of values. You must specify a minimum and a maximum value for each range. For example, if you specify MAC Source Address as a filtering field, you must specify which address(es) to filter. You could specify 0x0000A2000001 as the minimum value and 0x0000A2000003 as the maximum value. Then, all incoming packets would be checked to see if their MAC Source Address field was between 0x0000A2000001 and 0x0000A2000003.

- Action

  An action defines what happens to an incoming packet that matches one of the ranges for every filtering field in the filter. The actions are:

  — High Queue

  Specifies that any frame that matches the filter will be queued to the high queue.

  — Low Queue

  Specifies that any frame that matches the filter will be queued to the low queue.

Note: Any frame that does not match a filter is automatically queued to the Normal Queue.

Data Link Header and IP Header Fields

When you create a content-based priority filter you have the option of using either pre-defined fields or user-defined fields. The subsequent sections describe:

- Pre-defined fields for the datalink header
- Pre-defined fields for the IP header
- User-defined fields
- How to specify a user-defined field, which is slightly different than specifying a pre-defined field.

Datalink Pre-Defined Fields

At the data link level, you can configure priority filters based on the header fields within each of the supported encapsulation methods; they are:

- Ethernet
- IEEE 802.2 logical link control
- IEEE 802.2 LLC with SNAP header
- Novell Proprietary

All frame headers include both a MAC Destination Address and a MAC Source Address field; therefore, filtering on these two fields is possible for each supported encapsulation method. Aside from that, each encapsulation method has specific pre-defined filtering fields, which are described in Table 17-2.

Table 17-2. Pre-Defined Filter Fields for Each Encapsulation Method

Encapsulation Method	Pre-Defined Fields
All	MAC Source Address MAC Destination Address
Ethernet	Ethernet Type
802.2	Length SSAP DSAP Control
SNAP	Length Protocol ID/Organization code Ethertype

Note: There are no additional filtering fields for Novell; it allows filtering only on the MAC Source and MAC Destination Address fields.

IP Pre-Defined Fields

At the IP level, you can configure priority filters based on the header fields in the IP header; they are:

- Type of Service
- IP Destination Address
- IP Source Address
- UDP Source Port
- UDP Destination Port
- TCP Source Port
- TCP Destination Port
- Protocol

User-Defined Fields

Protocol prioritization provides the ability to prioritize traffic based upon specified bit pattern(s) contained within either the datalink or IP header. When creating a priority filter on user-defined fields, you specify the Reference, Offset, and Length, that together describe the location of the field on the incoming packet.

- Reference

  Positions the filtered bit pattern within the incoming frame. There are two reference points: the first is at the beginning of the Datalink header, and the second is at the beginning of the IP header.

- Offset

  Positions the filtered bit pattern (measured in bits), within either the data link or IP level header.

- Length

  Specifies the bit-length of the filtered field.

After specifying the Reference, Offset, and Length of your field, you specify one or more ranges for that field. For more information, see *Specifying User-Defined Fields* later in this chapter. Table 17-3 shows Reference, Offset, and Length for datalink header filtering fields. Table 17-4 shows Reference, Offset and Length for IP header filtering fields.

Table 17-3. Reference, Offset, and Length of Datalink Header Filtering Fields

Field	Reference	Offset	Length
MAC Destination Address	MAC	0	48
MAC Source Address	MAC	48	48
Ethernet Type	MAC	96	16
802.2 Length	MAC	96	16
802.2 DSAP	DATA_LINK	0	8
802.2 SSAP	DATA_LINK	8	8
802.2 Control	DATA_LINK	16	8
SNAP Length	MAC	96	16
SNAP Protocol ID/ Organization Code	DATA_LINK	24	24
SNAP Ethertype	DATA_LINK	48	16

Table 17-4. Reference, Offset, and Length of IP Header Filtering Fields

Field	Reference	Offset	Length
Type of Service	HEADER_START	8	8
Protocol	HEADER_START	72	8
Source IP Address	HEADER_START	96	32
Destination IP Address	HEADER_START	128	32
UDP/TCP Source Port	HEADER_END	0	16
UDP/TCP Destination Port	HEADER_END	16	16

Specifying User-Defined Fields

When you use the Configuration Manager to create or edit a template (described in the next section), you may add or edit filtering fields. When you access the appropriate menu to add a field, each pre-defined filtering field is represented as an option in that menu. The one other option is the User-Defined option. The User-Defined option allows you to set up specialized filtering fields based on bit pattern(s) within the Datalink or IP header.

Setting up user-defined fields is similar to setting up pre-defined fields, except that you must specify the field's location within the packet. (With pre-defined fields, you do not have to do that; their locations are established.) So, essentially, there is one extra step (window) required to specify a user-defined field.

When you select the User-Defined option, the User-Defined Filter Field Window appears. In this window, you specify the field's location within the header. To do this, you must set the field's Reference, Offset, and Length. Then, you specify a range associated with the bit field described by the Reference, Offset, and Length.

For example, suppose that you are bridging VINES traffic over Ethernet, and you want all packets with a destination network number of 1234 (hex) to take precedence over all other traffic, you would set up filtering fields as follows:

1. Specify an Ethernet Type field of 0xBAD (VINES). Ethernet Type is a pre-defined field.
2. Determine the Reference, Offset, and Length values of the Destination Network field within the header (see Figure 17-8).

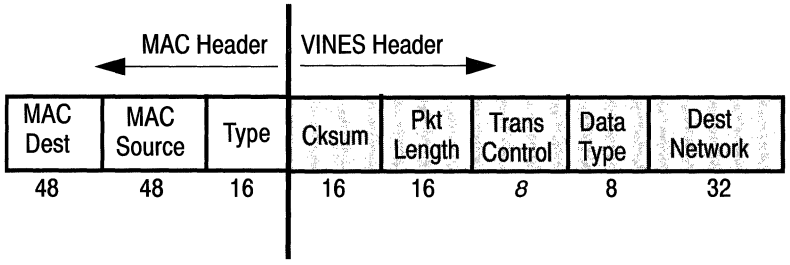


Figure 17-8. VINES Header

3. Set the Reference, Offset, and Length in the Add User-Defined Field Window, as follows (see Figure 17-9).
   - — Reference = MAC (beginning of frame)
   - — Offset = 160 bits (sum of all fields that precede the Destination Network field, or 48+48+16+16+16+8+8)
   - — Length = 32 bits

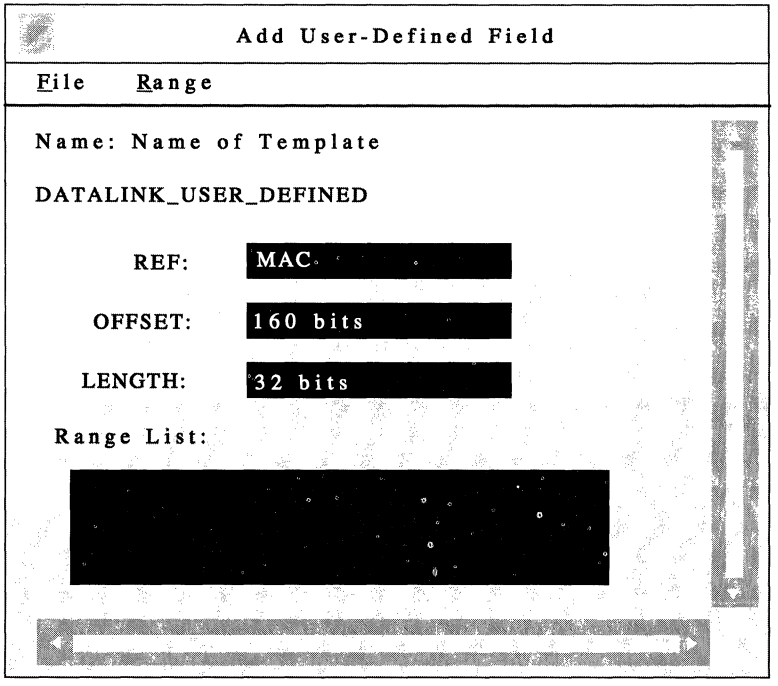


Figure 17-9. Add User-Defined Field Window

4. Specify the range to go with the field described by Reference, Offset, and Length.

   You specify the range the same way that you specify a range for a pre-defined filtering field; you simply select the Range/Add Range option. Then, you enter a minimum and maximum value in the appropriate boxes. In this case, you would specify 0x1234 for both the minimum and maximum values.

The procedures to add, delete, and edit ranges for a user-defined field are the same as the procedures for a pre-defined field. They are described in *Adding Ranges, Deleting Ranges*, and *Modifying Ranges*.

Implementation Notes

This section provides you with some suggestions about ways to use Protocol Prioritization. Aside from prioritizing LAT and Telnet traffic, you might also find Protocol Prioritization useful for prioritizing routing protocols such as RIP or OSPF. If you are running the Spanning Tree Protocol, you could prioritize your Spanning Tree traffic. You may also want to prioritize Source Routed Bridge traffic.

Provided below are the Fields, Ranges and Actions necessary to create content-based priority filters to prioritize each of these types of traffic.

Prioritizing LAT Traffic

To prioritize your LAT traffic, create a content-based priority filter with the following information:

- Field(s): Datalink, Ethernet Type
- Range(s): 0x6004 to 0x6004
- Action: High Queue

Prioritizing Telnet Traffic

To prioritize your Telnet traffic, create a content-based priority filter with the following information:

- Field(s): IP, TCP Destination Port
- Range(s): 23 to 23
- Action: High Queue

Prioritizing RIP Traffic

To prioritize your RIP traffic, create a content-based priority filter with the following information:

- Field(s): IP, UDP Destination Port
- Range(s): 520 to 520
- Action: High Queue

Prioritizing OSPF Traffic

To prioritize your OSPF traffic, create a content-based priority filter with the following information:

- Field(s): IP, Protocol Type
- Range(s): 89 to 89
- Action: High Queue

Prioritizing Spanning Tree Traffic

To prioritize your Spanning Tree traffic, create a content-based priority filter with the following information:

- Field(s): Datalink, DSAP/ SSAP/ Control
- Range(s): 0x42 to 0x42/0x42 to 0x42/0x03 to 0x03
- Action: High Queue

Prioritizing Native Source Routed Bridge Traffic

To prioritize your native SRB traffic, create a content-based priority filter with the following information:

- Field(s): SNAP, Ethertype
- Range(s): 0x8101 to 0x8101
- Action: High Queue

Prioritizing IP Encapsulated Source Routed Bridge Traffic

To prioritize your IP encapsulated SRB traffic, create a content-based priority filter with the following information:

- Field(s): IP, UDP Destination Port
- Range(s): 12288 to 12288
- Action: High Queue

Using the Configuration Manager to Configure Filters

The following sections assume that you are familiar with the datalink and IP header fields, and actions, and with setting up a user-defined field if you intend to do so. They explain how to use the Configuration Manager to configure priority filters, which includes:

- Adding a content-based priority filter to an interface, which includes creating a template
- Deleting a content-based priority filter
- Editing templates
  - Copying a template
  - Editing a template (its fields, ranges, and actions)
  - Deleting a template
- Editing a content-based priority filter (its fields, ranges, and actions)
- Editing a length-based priority filter
- Editing Protocol Prioritization interface-specific parameters

Note: As stated before, length-based filters can only be created at the circuit level. If you want to create a length-based priority filter, refer to *Configuring Circuits*.

Adding a Content-Based Priority Filter to an Interface

This section describes adding a content-based priority filter to an interface when there is no existing template that suits your needs (in this case, a template must be created). Before you can add a content-based priority filter to an interface, you must have already added Protocol Priority to the circuit, described in *Configuring Circuits*. If you have not done this, you will have to go back through the circuit level and add Protocol Priority to the appropriate circuits.

The process described in the following steps involves: naming the template, adding filtering fields and ranges to the template, adding an

action to the template, then applying (saving) the template to the appropriate interface. Start at the Wellfleet Configuration Manager Window, and complete the following steps:

1. Select the Protocols/Protocol Priority/Content Based option.

   The Filters Window appears (see Figure 17-10). This window lists all the interfaces to which a content-based priority filter has been added. In this example, no content-based filters have yet been added to any interface.

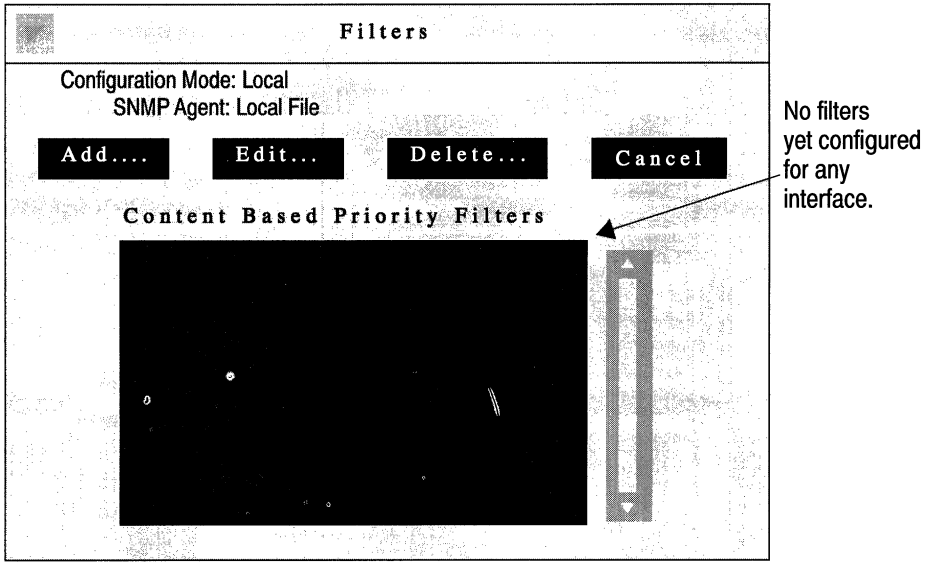


Figure 17-10. Filters Window

2. Click the Add button.

   The Add Filter Window appears (see Figure 17-11).

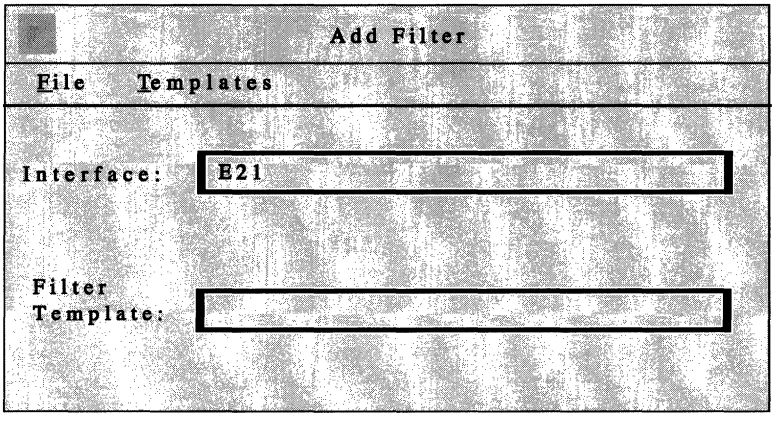


Figure 17-11. Add Filter Window

3. Select the Templates/Add Template option.

   The Edit Filter Window appears with all its fields blank (see Figure 17-12).

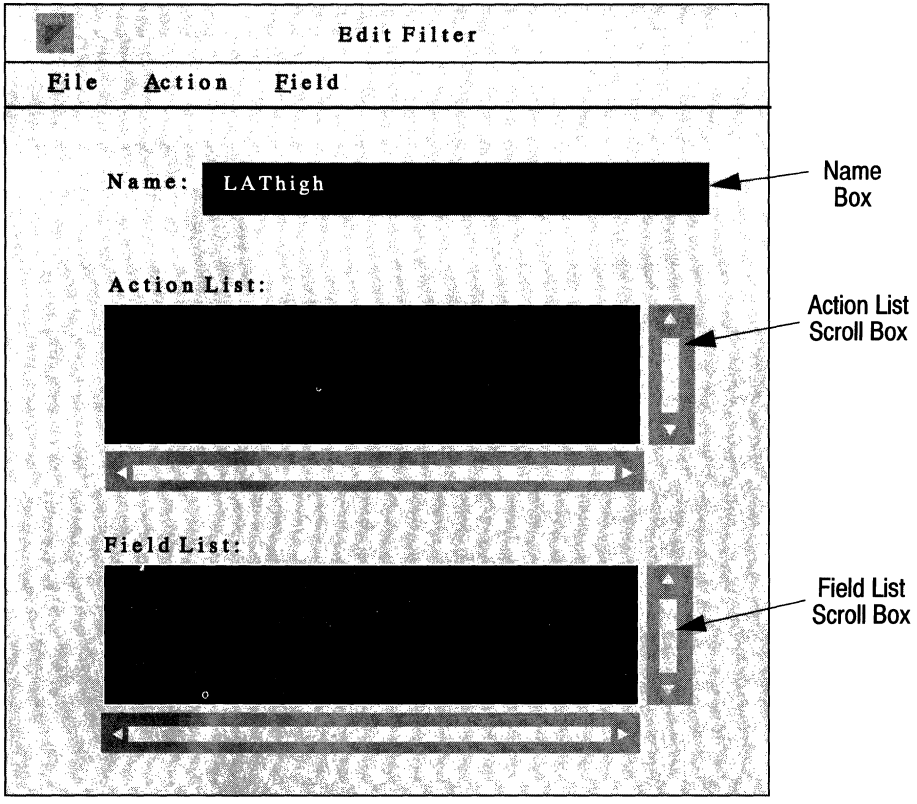


Figure 17-12. Edit Filter Window

4. Enter a name for this new template in the Name box.

   It is best to give descriptive names to your templates. For instance, in this example, the template will be called *LAThigh* because it will contain information which will direct all LAT traffic into the high queue.

   Next, you must add filtering fields and ranges to the template.

5. Select the Field/Add Field option, then select either Datalink or IP (Datalink is appropriate in this example).

Another menu appears showing you the header-specific filtering field options.

6. Select the field on which you wish to filter packets.

   In this example, the Ethertype field is chosen by selecting the Add Field/Datalink/DataLink/Ethernet Type option.

   The Edit Field Window now appears (see Figure 17-13). For any field you choose, you need to specify at least one associated range.

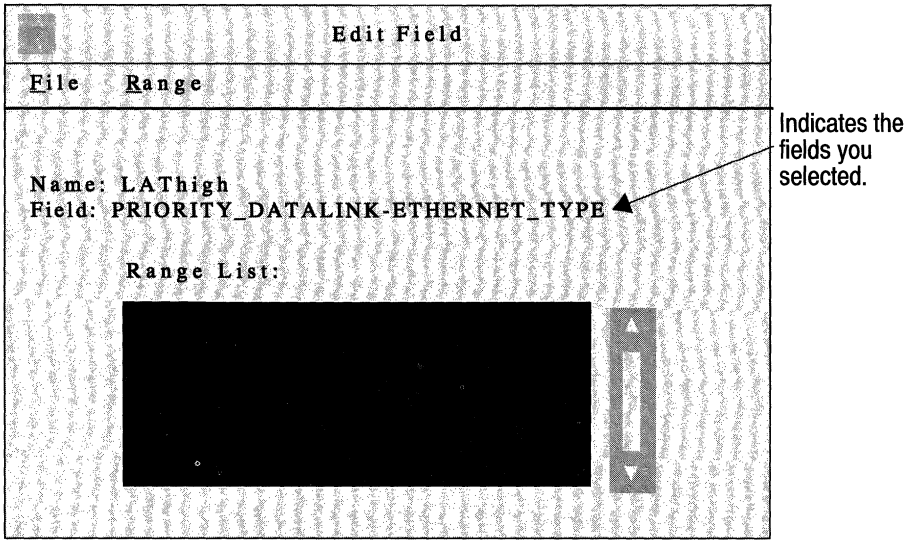


Figure 17-13. Edit Field Window

7. Select the Range/Add Range option.

   The Edit Range Window appears (see Figure 17-14).

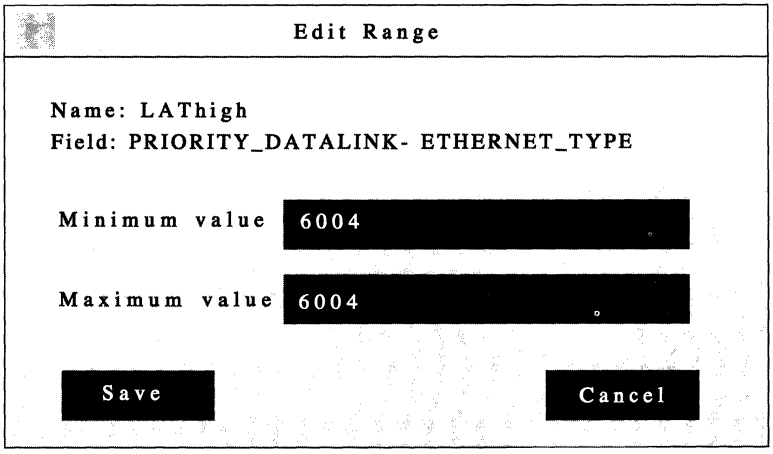


Figure 17-14. Edit Range Window

8. Specify the low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that value in both boxes. In this case, the Ethertype 6004 was specified as the minimum value, and also as the maximum value. Each incoming packet will be checked to see if its Ethertype is equal to 6004 (which indicates it is DEC LAT traffic).

   The range you just specified will now appear in the Range List scroll box in the Edit Field Window. You can add more ranges to a field by repeating steps 8 and 9.

9. When you are finished adding ranges to this field, select the File/Save option.

   You are returned to the Edit Filter Window.

   You can add additional filtering fields to this template; simply follow steps 6 through 10. Each field you add will appear in the Field List scroll box on the Edit Filter Window.

Next, you must add an action to your template.

10. From the Action menu, select either Datalink or IP (datalink is appropriate in this example); then, select the Add Action and either High Queue or Low Queue.

    In this example, High Queue is chosen by selecting the Action/Datalink/Add Action/High Queue option.

11. Select the File/Save option.

    This template is saved, and now appears in the Filter Template box in the Add Filter Window.

    Finally, you apply this template to an interface.

12. In the Add Filter Window, make sure the appropriate template appears in the Filter Template box; then, make sure the interface to which you want to apply it appears in the Interface box (see Figure 17-15).

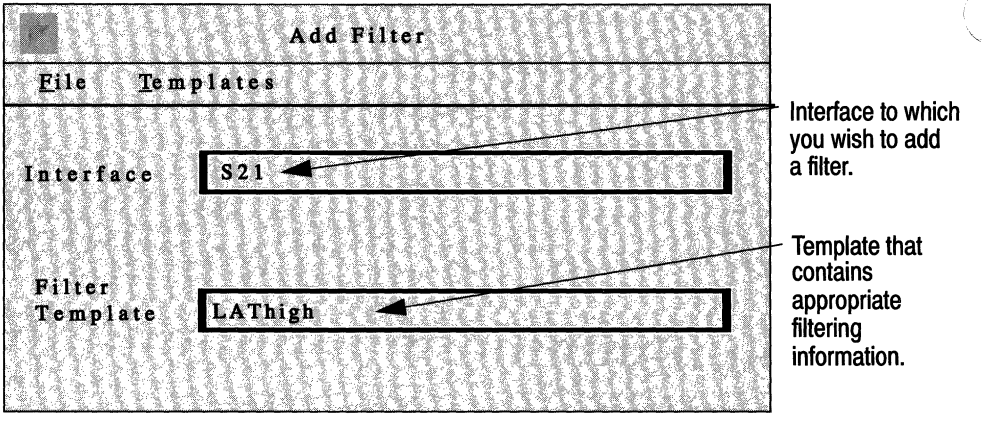


Figure 17-15. Specifying the Appropriate Template and Interface

13. When the appropriate template and interface are specified, select the File/Save option.

In this example, the template called *LAThigh* is being applied to interface S21. The fields specified in this template now serve as a filter on interface S21. That is, each of S21's incoming packets will be checked to see if it matches any of *LAThigh's* fields and associated ranges. If a match is found, the action (High Queue, in this case) will be imposed on the packet; it will be queued to the high queue.

You are now returned to the Filters Window. The filter you just created for interface S21 is listed in the Content Based Priority Filters scroll box. Assuming this is the first filter added to an interface, the window will appear as shown in Figure 17-16.

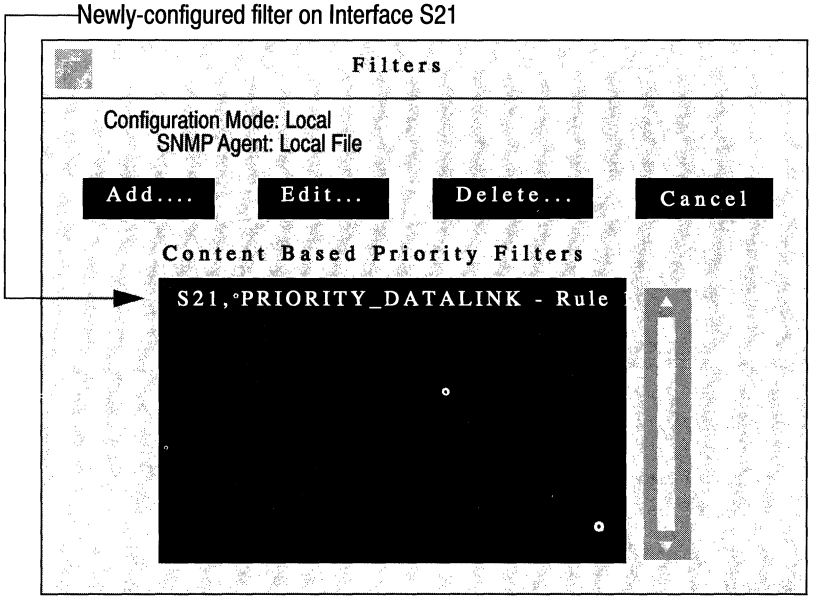


Figure 17-16. Content-Based Priority Filter in Filters Window

Deleting a Content-Based Priority Filter

If you want, you can delete a content-based priority filter from an interface. When you do, it affects only the interface from which the filter is removed. To delete a content-based priority filter from an interface, start at the Wellfleet Configuration Manager Window and complete the following steps:

1. Select the Protocols/Protocol Priority/Content Based option.

   The Filters Window appears listing all of the interfaces having content-based priority filters (see Figure 17-17).

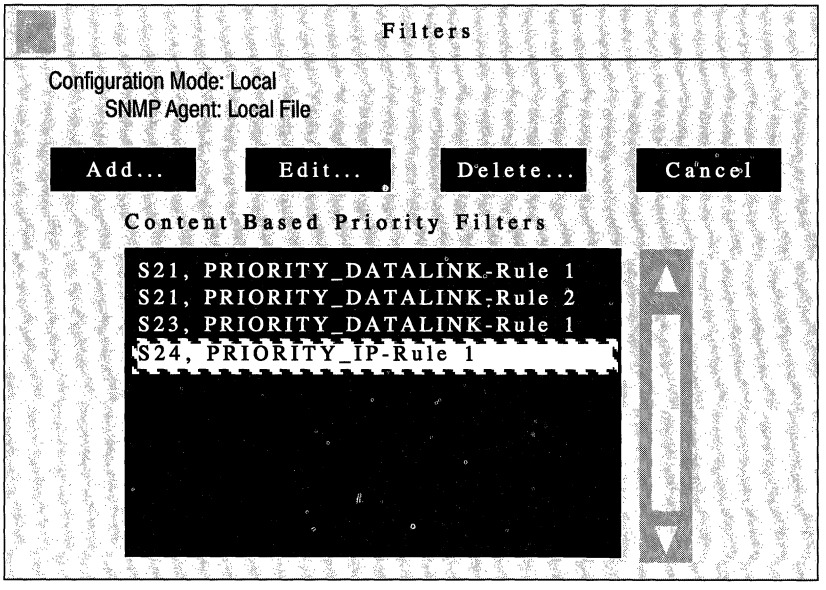


Figure 17-17. Filters Window

2. Select the appropriate interface/priority filter pair.

   In this example, S24,PRIORITY_IP-Rule 1 is chosen, indicating that the only priority filter on interface S24 is to be deleted.

3. Click the Delete button.

The Delete Filter Window appears. The interface/priority filter pair you selected in step 2 appears in the Delete Filter box.

4. Click the Delete button.

The filter is deleted from the interface and no longer appears in the Content Based Priority Filters scroll box on the Filters Window.

Editing Templates

When you want to add a content-based priority filter to an interface, you do not always have to create a new template. More often than not, you will be able to use existing templates to build new ones. For instance, suppose that on certain interfaces you want all frames with a MAC Source Address of 0x00A2000025 to take precedence over other traffic. Suppose also that there is a template named *Src20high* that contains fields and actions instructing all frames with a MAC Source Address of 0x00A2000020 to be queued to the High Queue. You can do one of two things:

- Specify the template (in this case *Src20high*), and edit it.

  Changes to this template will *not* affect interfaces to which this template has already been applied.

- Copy the template, rename, and edit it.

  This creates an entirely new template (with the same fields, ranges and actions as those in the *Src20High* template), that you can rename and edit to suit your needs.

These two options are discussed in the next two sections: *Copying a Template*, and *Specifying a Template.*

Note: You may also edit any template using a text editor. All templates are stored in a file called *template.flt*.

Copying a Template

Copying a template to a new name before you edit it is sometimes favorable, especially if you need to preserve the original template. To copy a template, complete the following steps. Then, proceed to the section entitled *Editing Fields, Ranges, and Actions* for instructions on editing this new template.

1. At the Wellfleet Configuration Manager Window, select the Protocols/ Protocol Priority/Content Based option.

   The Filters Window appears.

2. Click the Add button; the Add Filter Window appears (see Figure 17-18).

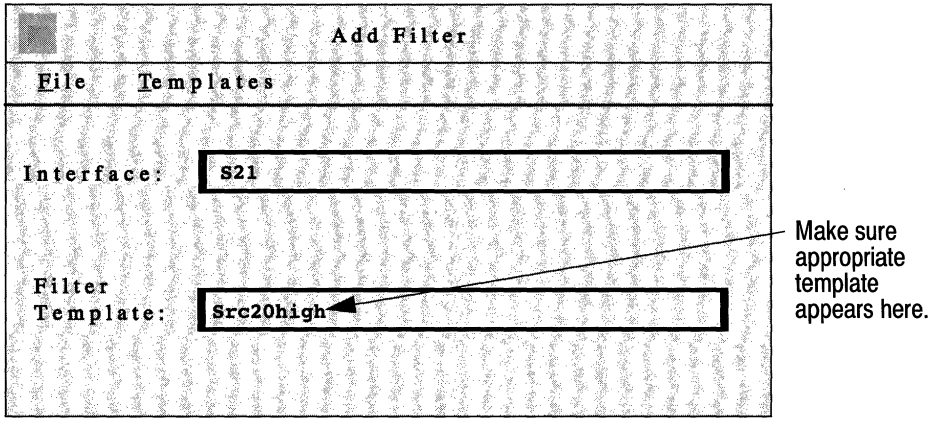


Figure 17-18. Add Filter Window

3. If the Filter Template box is displaying the name of the template you wish to copy, go to the next step. If the Filter Template is *not* currently displaying the name of the template you wish to copy, click on the box. A menu displaying all existing templates appears; choose the template you wish to copy.

4. Select the Templates/Copy Template option.

   The Copy Filter Template Window appears; see Figure 17-19.

5. Enter a name for the new template in the box provided.

   Remember that it is a good idea to give your template a name that reflects it contents.

6. Click the Save button.

   You are returned to the Add Filter Window. The name you just assigned to the new template appears in the Template Filter box.

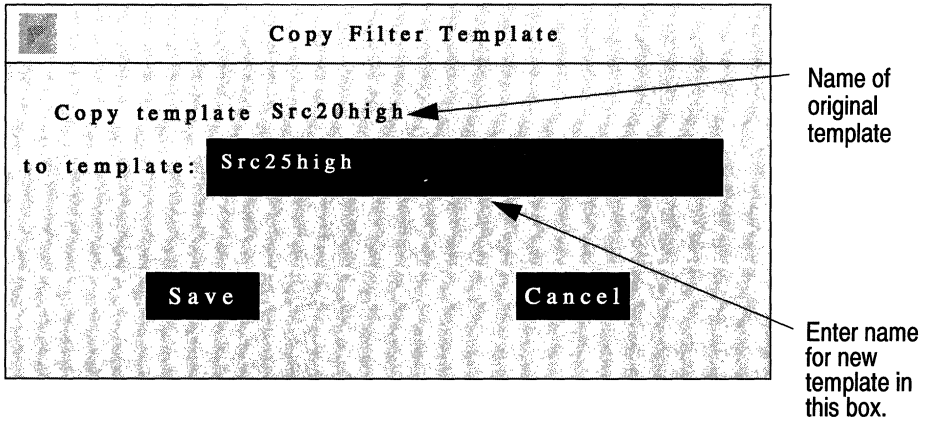


Figure 17-19. Copy Filter Template Window

7. Select the Templates/Edit Template option.

   The Edit Filter Window for the new template appears.

For instructions on editing this template, proceed to the section entitled *Editing Fields, Ranges and Actions*.

Specifying a Template

If you do not want or need to preserve the original template, you can simply edit it without first copying and renaming it. You simply need to specify it; start at the Wellfleet Configuration Manager Window, and complete the following steps:

1. Select the Protocols/Protocol Priority/Content Based option.

   The Filters Window appears.

2. Click the Add button; the Add Filter Window appears (see Figure 17-20).

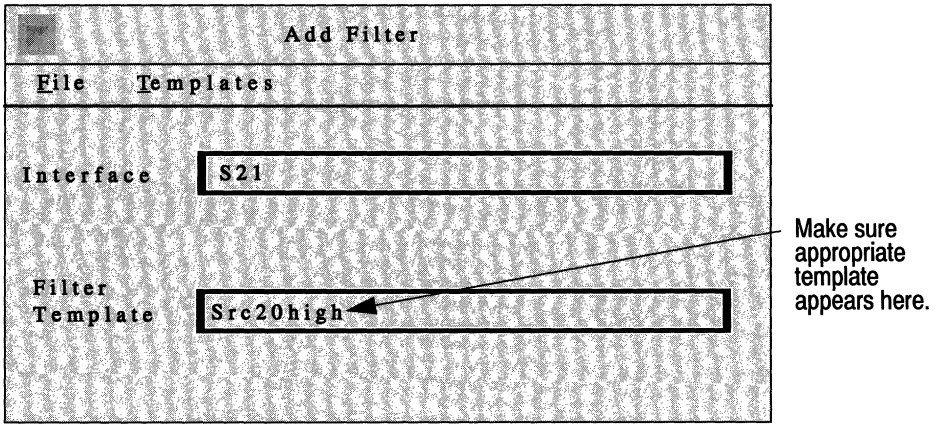


Figure 17-20. Add Filter Window

3. If the Filter Template box displays the name of the template you wish to edit, go to the next step. If the Filter Template box does not display the name of the template you wish to edit, click on the box. A menu appears listing all existing templates; choose the appropriate one.
4. Select the Templates/Edit Template option.

The Edit Filter Window for this template appears (see Figure 17-21). In this case, the template called *Src20high* was chosen.

For instructions on editing this template, proceed to the next section, *Editing Fields, Ranges, and Actions*.

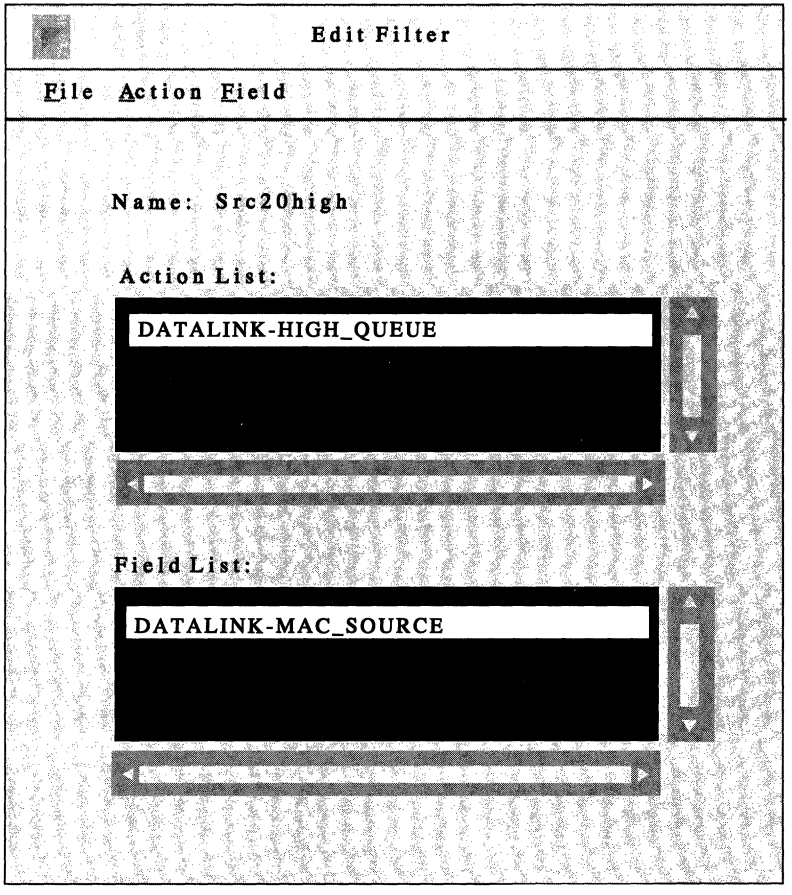


Figure 17-21. Edit Filter Window for a Specific Template

Editing Fields, Ranges, and Actions

Once you have either copied or specified a template, you can edit its fields, ranges, and actions. You have the following options, which are described in subsequent sections:

- Deleting or adding filtering fields
- Deleting, adding or modifying field ranges
- Deleting or adding actions

Deleting a Field

If you no longer want a field to be included in a template, follow these steps to remove it:

1. From the Field List scroll box, select the field you wish to delete.
2. Select the Field/Delete Field option.

   The Delete Field Window appears; the field you selected to delete appears in the Delete Field box.
3. Click the Delete button.

   You are returned to the Edit Filter Window. The field you just deleted no longer appears in the Field List scroll box.

Adding a Field

If you want to add a field to a template, complete the following steps. If you intend to add user-defined fields, refer to *Setting Up User-Defined Fields* (earlier in this chapter), which explains the special considerations of specifying user-defined fields.

1. Select the Field/Add Field option, then select either the Datalink or the IP option.

   Another menu appears showing you the header-specific filtering field options.
2. Select the field you wish to add to this template.

   The Edit Field Window appears (see Figure 17-22). For any field you choose, you must specify at least one range.

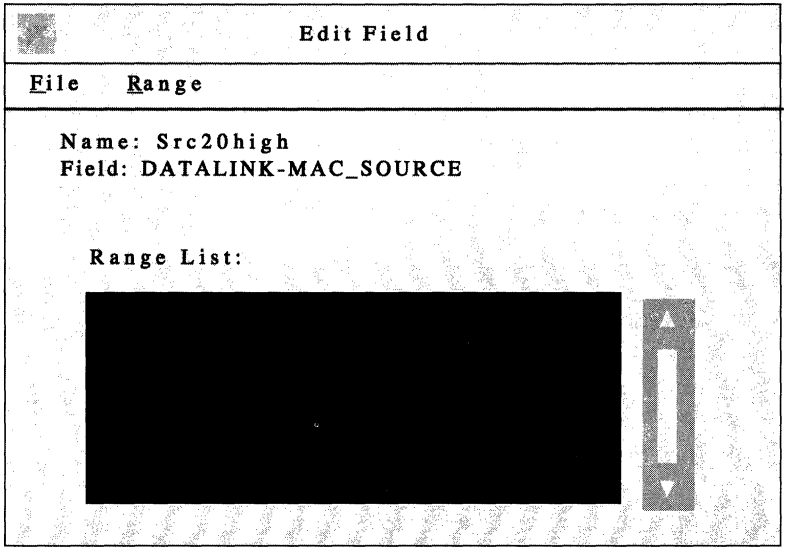


Figure 17-22. Edit Field Window

3. Select the Range/Add Range option.

   The Edit Range Window appears (see Figure 17-23).

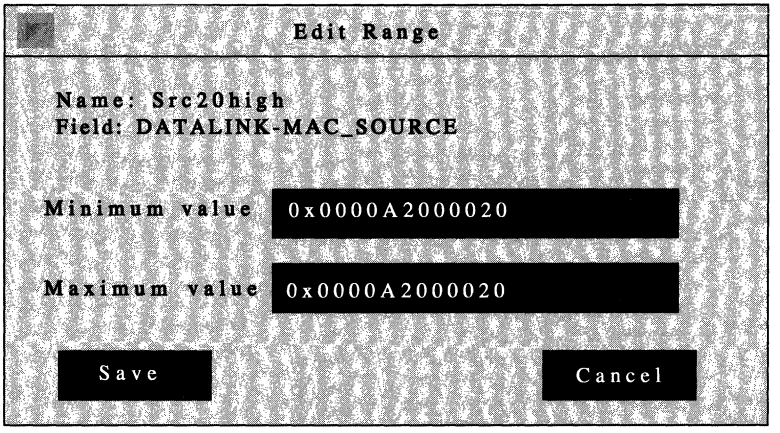


Figure 17-23. Edit Range Window

4. Specify the low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that same value in both boxes. In this case, the MAC Source Address 0x0000A2000020 was specified as both the minimum and maximum value. Each incoming packet will be checked to see if its MAC Source Address equals 0x0000A2000020.

   The range you just specified now appears in the Range List scroll box in the Edit Field Window. You can add more ranges to this field by repeating steps 3 and 4 for each range you wish to add.

5. When you are finished adding ranges to this field, select the File/Save option.

   You are returned to the Edit Filter Window.

You can add filtering fields to this template. Simply follow steps 1 through 5 for each new field you wish to add. Each field you specify will appear in the Field List scroll box on the Edit Filter Window.

Deleting Ranges

If you need to delete a range from a template's field, complete the following steps.

1. From the Field List scroll box, select the field from which you wish to delete a range.
2. Select the Field/Edit Field option.

   The Edit Field Window appears (see Figure 17-24). It lists all of the ranges associated with this field.

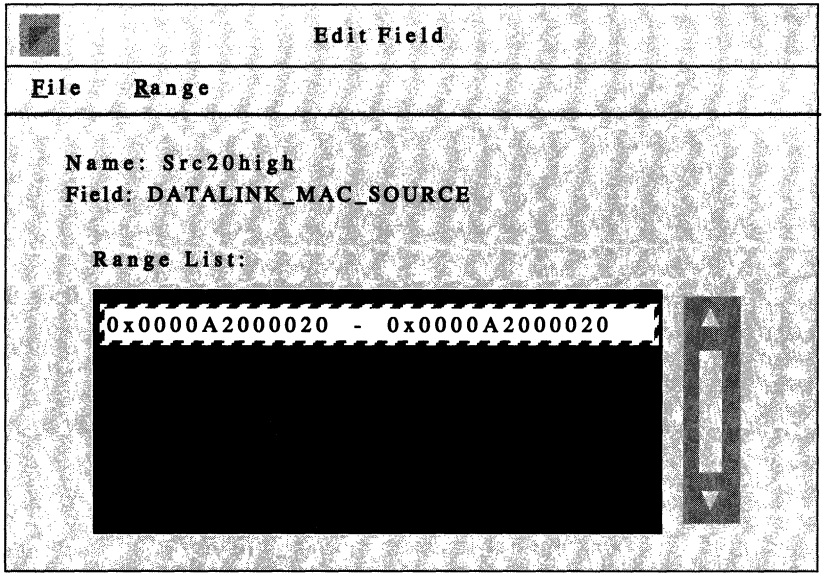


Figure 17-24. Edit Field Window

3. Select the range you wish to delete from this field.

   In this example, the range 0x0000A2000020 - 0x0000A2000020 is selected.
4. Select the Range/Delete Range option.

   The Delete Range Window appears.

5. Click the Delete button.

   You are returned to the Edit Field Window. The range you just deleted from this field no longer appears in the Range List scroll box.

Adding Ranges

If you need to add a range to a template's field, complete the following steps.

1. From the Field List scroll box, select the field to which you wish to add a range.
2. Select the Field/Edit Field option.

   The Edit Field Window appears (see Figure 17-25). It lists all of the ranges associated with this field.

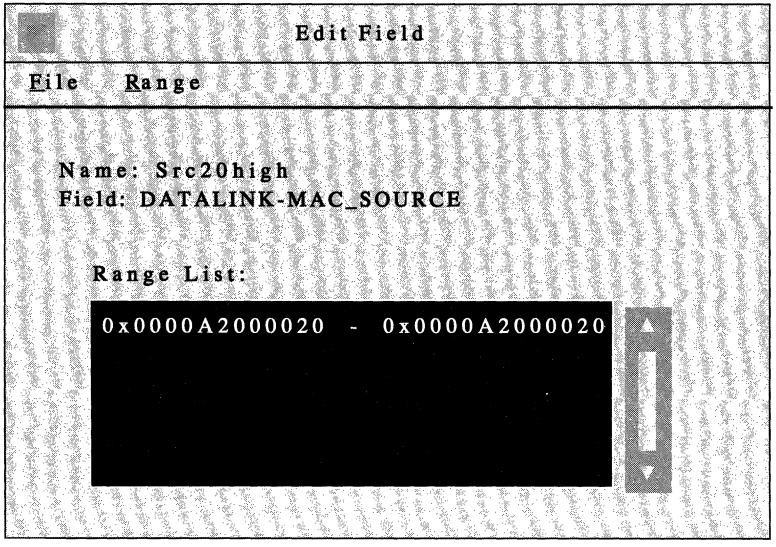


Figure 17-25. Edit Field Window

3. Select the Range/Add Range option.

   The Edit Range Window now appears (see Figure 17-26).

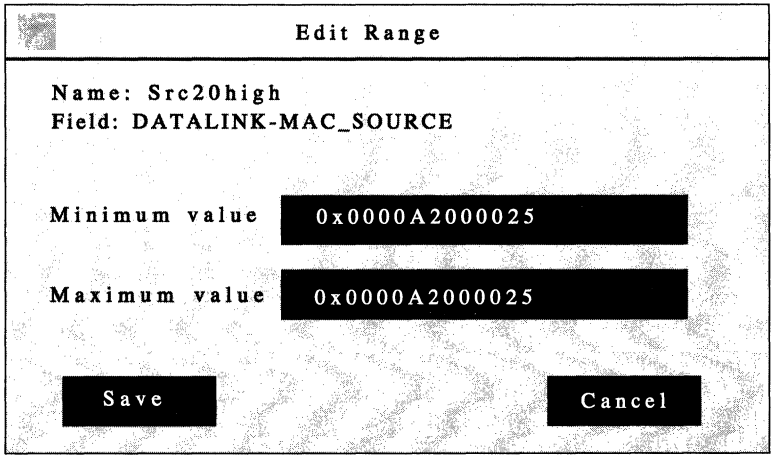


Figure 17-26. Edit Range Window

4. Specify the low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that same value in both boxes. In this case, the MAC Source Address 0x0000A2000025 was specified as both the minimum and maximum value. Each incoming packet will be checked to see if its MAC Source Address equals 0x0000A2000025.

   The range you just specified now appears in the Range List scroll box in the Edit Field Window. You can add more ranges to this field by repeating steps 3 and 4 for each range you want to add.

5. When you are finished adding ranges to this field, select the Field/Save option.

   You are returned to the Edit Filter Window.

Modifying Ranges

If you need to change a field's range, complete the following steps.

1. Select the field for which you wish to modify a range from the Field List scroll box.
2. Select the Field/Edit Field option.

   The Edit Field Window appears (see Figure 17-27). It lists all of the ranges that have been specified for this field.

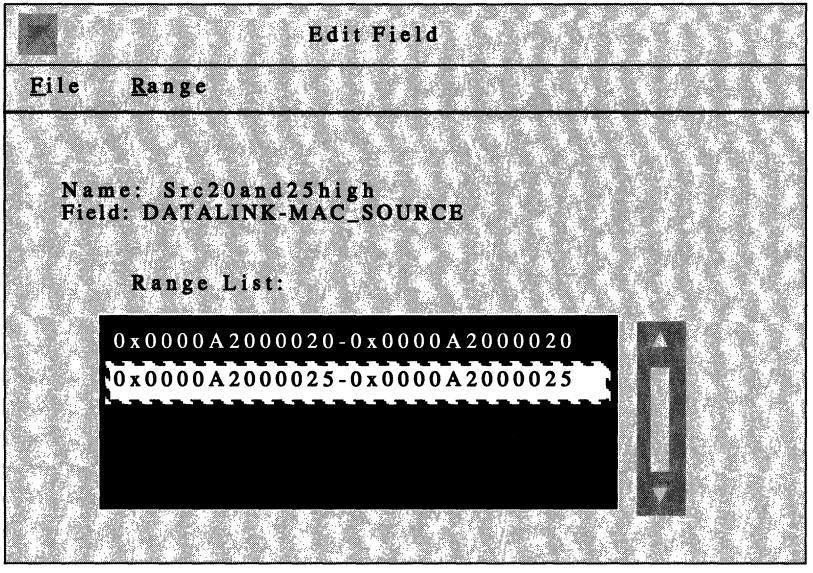


Figure 17-27. Edit Field Window

3. Select the range you wish to modify.

   In this example, range 0x0000A2000025-0x0000A2000025 is selected.
4. Select the Range/Edit Range option.

   The Edit Range Window appears (see Figure 17-28).

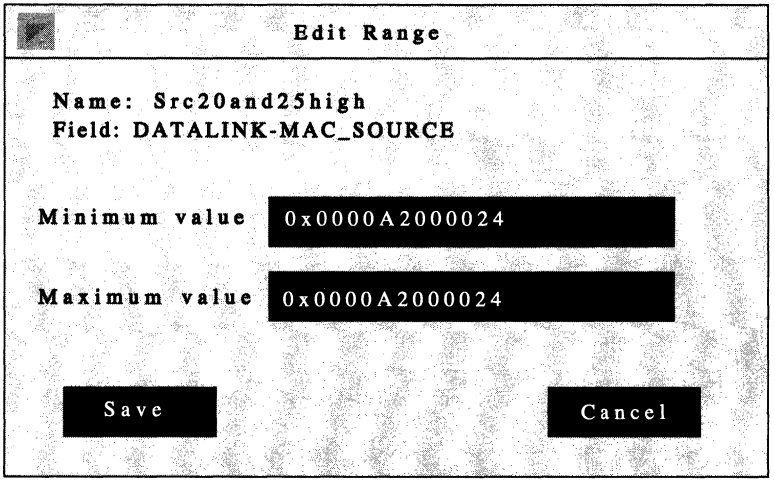


Figure 17-28. Edit Range Window

5. Specify the new low and high ends of the range you want to filter in the Minimum value and Maximum value boxes, then click the Save button.

   If the range you want to filter consists of just one value, specify that same value in both boxes. In this case, the MAC Source Address 0x0000A2000024 was specified as both the minimum and maximum value. Each incoming packet will be checked to see if its MAC Source Address equals 0x0000A2000024.

   The range you just specified now appears in the Range List scroll box in the Edit Field Window. For each range you want to modify, repeat steps 3 through 5.

6. When you are finished modifying ranges for this field, select the Field/Save option.

   You are returned to the Edit Filter Window.

Deleting Actions

If you no longer want an action to be included in a template, follow these steps to remove it:

1. From the Action menu, select the either the Datalink or the IP option, then select Delete Action.

   The Delete Traffic Filter Action Window appears.

2. Click the Delete button.

   You are returned to the Edit Filter Window. The action you have just deleted no longer appears in the Action List scroll box. You will have to add an action for this template to be complete.

Adding Actions

If you want to add an action to a template, follow these steps:

1. From the Action menu, select either the Datalink or the IP option; then select the Add Action option.
2. Select the action you wish to impose on packets that match any of the template's filtering fields, either High Queue or Low Queue.

   You are returned to the Edit Filter Window. The action you have just added appears in the Action List scroll box.

Deleting Templates

If you want to delete a template from your list of templates, begin at the Wellfleet Configuration Manager Window, and complete the following steps.

1. Select the Protocols/Protocol Priority/Content Based option.

   The Filters Window appears.

2. Click the Add button.

   The Add Filter Window appears (see Figure 17-29).

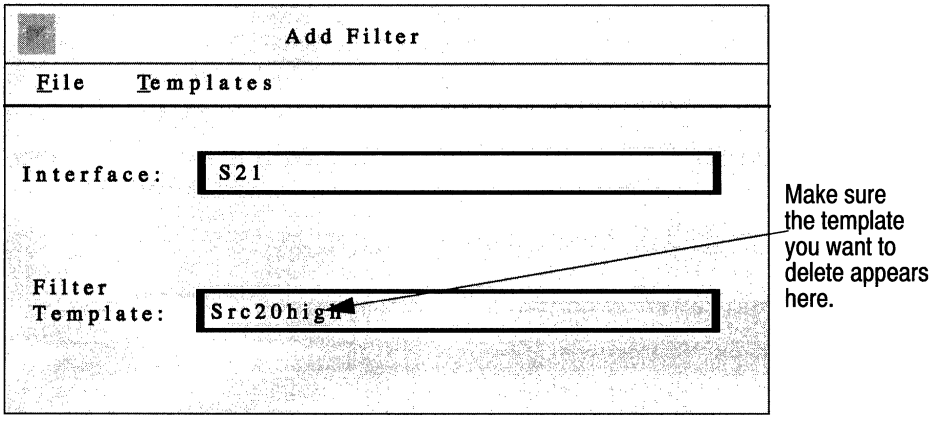


Figure 17-29. Add Filter Window

3. If the template that you wish to delete is displayed in the Filter Template box, go to the next step. If the template you wish to delete is not displayed in the Filter Template box, click on the box. A menu appears listing all existing templates; choose the template you wish to delete.
4. Select the Templates/Delete Template option.

   The Delete Filter Template Window appears.
5. Click the Delete button.

   You are returned to the Add Filter Window. The template you just deleted is no longer available in the Filter Template box.

Editing a Content-Based Priority Filter

If you want, you can edit content-based priority filters on individual interfaces. When you do, only the filter on that specific interface is affected. To edit a filter, start at the Wellfleet Configuration Manager Window and follow these steps.

1. Select the Protocol/Protocol Priority/Content Based option.

   The Filters Window appears listing all of the interfaces having content-based priority filters (see Figure 17-30).

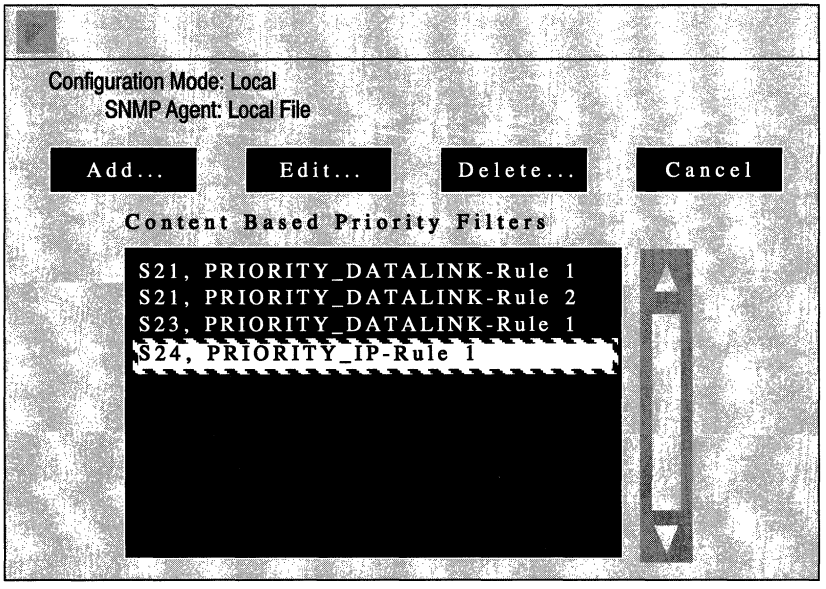


Figure 17-30. Filters Window

2. Select the appropriate interface/priority filter pair.

   In this example, S24,PRIORITY_IP-Rule 1 is chosen, indicating that the priority filter on interface S24 is to be edited.

3. Click the Edit button.

The Edit Filter Window for this interface/priority filter appears. Editing a content-based priority filter's fields, ranges and actions is the same as editing a template's fields, ranges, and actions. However, when you save your changes, it will affect the filter on this interface only. Refer to the section entitled *Editing Fields, Ranges, and Actions* for instructions on how to edit this filter.

Editing a Length-Based Priority Filter

Although you cannot add or delete length-based priority filters at the protocol level, you can edit them on a per filter basis. When a length-based priority filter is created, the filter automatically takes certain default values; you can edit these default values. When you do, only the length-based filter on the specific interface that you select is affected.

To edit a length-based filter, start at the Wellfleet Configuration Manager Window and follow these steps.

1. Select the Protocol/Protocol Priority/Length Based option.

   The Length Based Priority Filters Window appears listing all of the interfaces having length-based priority filters (see Figure 17-31).

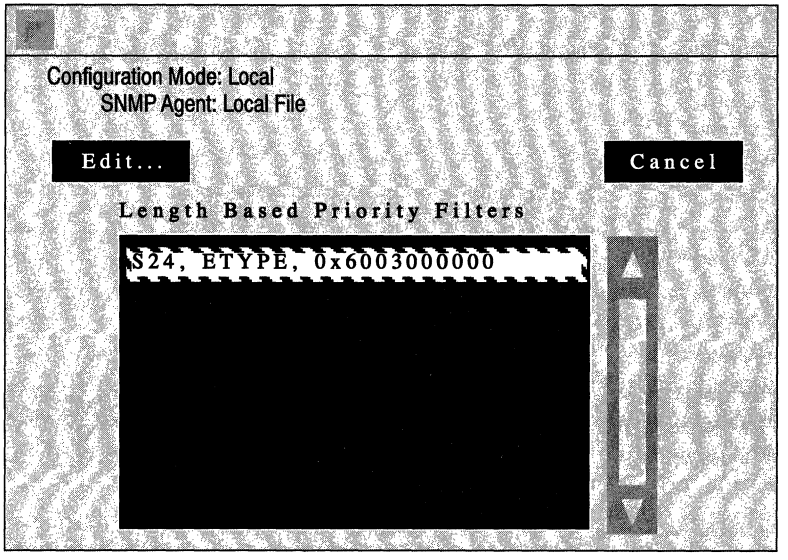


Figure 17-31. Length Based Priority Filters Window

2. Select the appropriate interface/priority filter pair.

   In this example, S24, ETYPE, 0x6003000000 is chosen, indicating that the length-based priority filter on interface S24 is to be edited.

3. Click the Edit button.

   The Edit Length Based Priority Filter Window appears (see Figure 17-32).

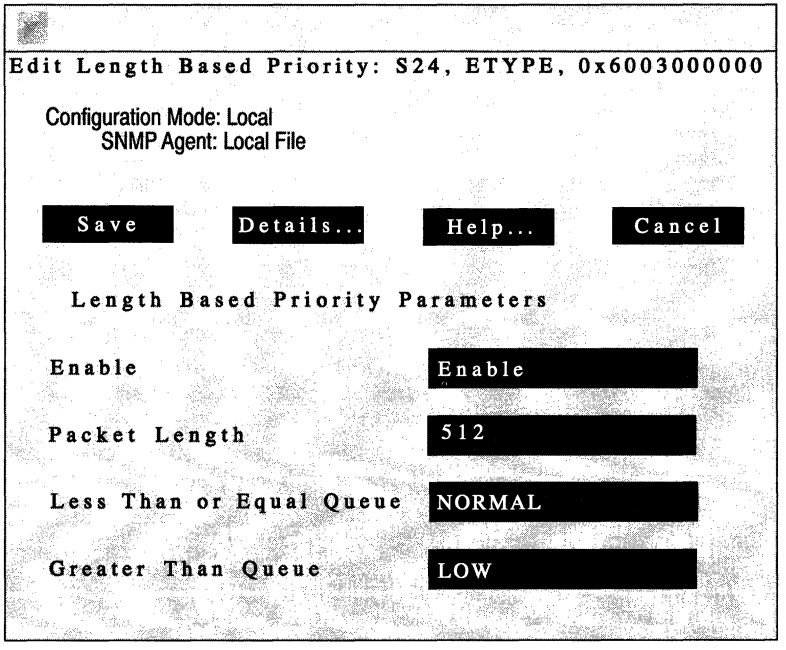


Figure 17-32. Edit Length Based Priority Filter Window

4. Edit those parameters you wish to change, then click the Save button to save your changes and exit this window.

   The length-based priority filter parameters are described below.

Parameter :	**Enable**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Toggles this length-based priority filter on and off on this interface. This parameter is useful if you want to temporarily disable a length-based priority filter rather than delete it.
Instructions:	Set this parameter to Disable if you want to disable this length-based priority filter. Or, set this parameter to Enable if you previously disabled this priority filter and now wish to reenable it.

Parameter :	**Packet Length**
Wellfleet Default:	Takes the Packet Length value that was supplied when this length-based priority filter was created.
Options:	Any packet length expressed in bytes
Function:	Defines a packet length measurement to which each packet is compared. An action is imposed on every packet depending on whether it is less than, equal to, or greater than Packet Length. This action also depends on the values of the Less Than or Equal Queue and the Greater Than Queue parameters.
Instructions:	Either accept the current Packet Length value, or enter a new Packet Length value in bytes.

Parameter:	**Less Than or Equal Queue**
Wellfleet Default:	Normal
Options:	Low, Normal, High
Function:	Dictates into which queue a packet will be queued if it's packet length is less than or equal to the value of Packet Length. That is, if Packet Length is 1024 bytes, any packet that is 1024 bytes or smaller will be queued to the queue you choose for this parameter.
Instructions:	Either accept the default, Normal, or select either Low or High.

Parameter:	**Greater Than Queue**
Wellfleet Default:	Low
Options:	Low, Normal, High
Function:	Dictates into which queue a packet will be queued if it's packet length is greater than the value of Packet Length. That is, if Packet Length is 1024 bytes, any packet that is 1025 bytes or larger will be queued to the queue you choose for this parameter.
Instructions:	Either accept the default, Low, or select either Normal or High

Editing Interface-Specific Protocol Prioritization Parameters

Any circuit to which you've added Protocol Prioritization takes certain default values that you can edit on a per interface basis at the protocol level. These interface-specific default values dictate how your length-based and/or content-based priority filters work on the interface. You can edit these parameters according to your network traffic needs. To do so, begin at the Wellfleet Configuration Manager Window, and complete the following steps:

1. Select the Protocols/Protocol Priority/Interfaces option.

   The Protocol Priority Interfaces Window appears (see Figure 17-33). This window shows all interfaces to which Protocol Prioritization has been added, regardless of whether or not there are any length-based or content-based priority filters currently active on the interfaces.

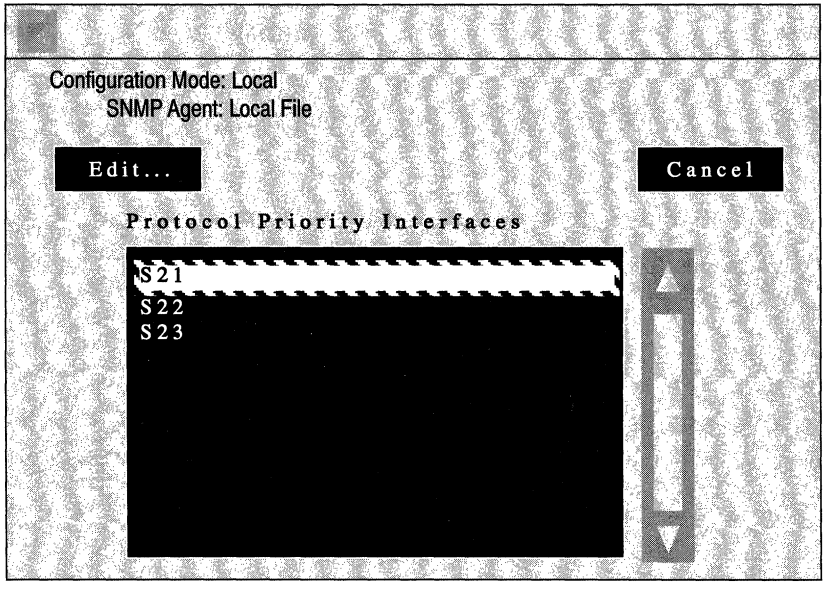


Figure 17-33. Protocol Priority Interfaces Window

2. Select the interface for which you wish to edit interface-specific parameters.
3. Click the Edit button.

   The Edit Protocol Priority Interface window appears for the interface that you selected (see Figure 17-34).

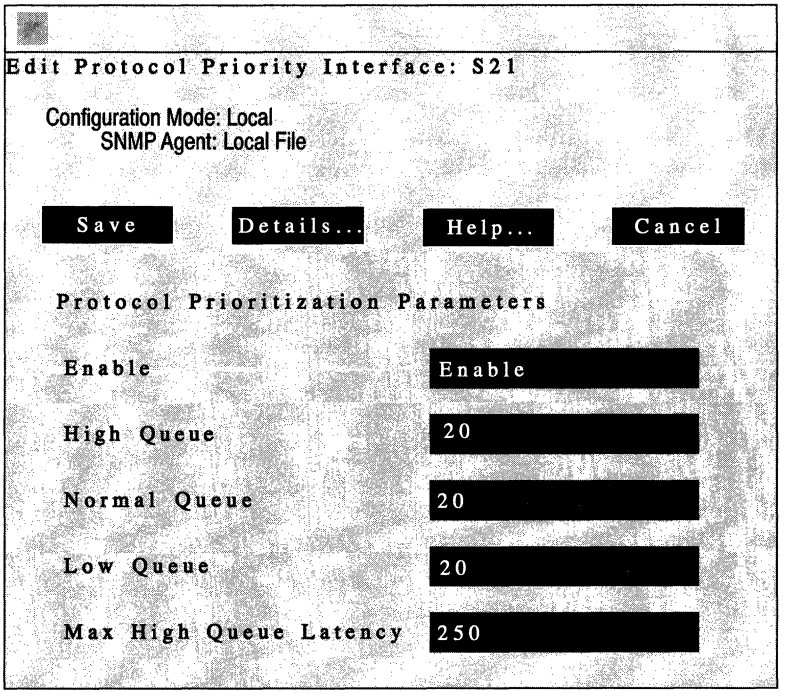


Figure 17-34. Edit Protocol Priority Interface Window

4. Edit those parameters you wish to change, then click the Save button to save your changes and exit this window.

   The interface-specific Protocol Prioritization parameters are described below.

Parameter:	**Enable**
Wellfleet Default:	Enable
Options:	Enable/Disable
Function:	Toggles Protocol Prioritization on and off on this interface. If you set this parameter to Disable, all priority filters, length-based and content-based will be disabled on this interface. This is useful if you temporarily want to disable all priority filters, rather than delete them.
Instructions:	Set to Disable if you want to temporarily disable all Protocol Prioritization activity on this interface. Or, set to Enable if you previously disabled Protocol Prioritization on this interface and now wish to reenable it.

Parameter:	**High Queue**
Wellfleet Default:	20 packets
Options:	0 to 63 packets
Function:	Dictates the size limit, in packets, of the High Priority Queue. For example, if the value of High Queue is 15, there can be no more than 15 packets in the High Priority Queue at any one time. For more information about how queue depths are used for tuning Protocol Prioritization in your network, see *Tuning Protocol Prioritization For Your Network.*
Instructions:	Either accept the default of 20 packets, or enter a new value.

Parameter:	**Normal Queue**
Wellfleet Default:	20 packets
Options:	0 to 63 packets
Function:	Dictates the size limit, in packets, of the Normal Priority Queue. For example, if the value of Normal Queue is 15, there can be no more than 15 packets in the Normal Priority Queue at any one time. For more information about how queue depths are used for tuning Protocol Prioritization in your network, see *Tuning Protocol Prioritization For Your Network.*
Instructions:	Either accept the default of 20 packets, or enter a new value.

Parameter:	**Low Queue**
Wellfleet Default:	20 packets
Options:	0 to 63 packets
Function:	Dictates the size limit, in packets, of the Low Priority Queue. For example, if the value of Low Queue is 15, there can be no more than 15 packets in the Low Priority Queue at any one time. For more information about how queue depths are used for tuning Protocol Prioritization in your network, see *Tuning Protocol Prioritization For Your Network.*
Instructions:	Either accept the default of 20 packets, or enter a new value.

Parameter :	**Max High Queue Latency**
Wellfleet Default:	250ms
Options:	100 to 5000ms
Function:	Indicates the greatest possible delay for your high priority traffic. It dictates how many normal or low priority bytes can be on the transmit queue at any one time, and therefore, the greatest delay that a high priority packet can experience. Latency is based on the line speed of the attached media. For a given line speed, the number of bits that can be queued to the transmit queue at any one time is determined by the configured latency value. For more information about how latency is used for tuning Protocol Prioritization in your network, see *Latency.*
Instructions:	Either accept the default latency of 250ms, or enter a new latency value.

Note: Wellfleet recommends accepting the default latency value of 250ms.

Chapter 18

Booting the Wellfleet Router with the Config File

List of Figures

Booting the Wellfleet Router with the Config File

About this Chapter

This chapter tells you how to use the Configuration Manager and the Remote File System Manager to boot the Wellfleet router with a configuration file.

A local or remote configuration file has no effect until you reboot the router using the configuration file. Dynamically made changes must be saved to a configuration file, with which the router must be booted, in order for them to have a permanent effect on the router. If you reboot the router without saving the dynamic changes to a configuration file, the changes are lost.

You reboot the router with a changed configuration file as follows:

- Save the configuration file locally.
- Use TFTP to transfer the configuration file to the router.
- Reboot the router with the configuration file.

Note: When you save a configuration file that was created in local mode, it is saved to the Site Manager workstation. Therefore, you must transfer the file to the router in order to reboot the router with it. When you save a configuration file that was created in remote mode, the file is transferred directly to the router. Use the procedures identified in this chapter when saving files in remote configuration mode or transferring files to the router to avoid corrupting the *config* file on the router.

If the router has a non-volatile file system (NVFS), you boot the router with the boot image (*boot.exe* in the BLN and BCN, or *ace.out* in the AFN) and a configuration file residing on a volume you specify. The volume in this case is the slot location of one file system on the router. Each router equipped with an NVFS is shipped with at least one volume. The default volume is displayed in the Boot and File Management Windows. These windows allow you to change the volume from which to boot.

If the router has a DOS file system, you boot from the volume named *A*.

Saving a Configuration File

The Configuration Manager does not create the configuration file until you save the configuration information you specified. The three sections that follow describe how to save a configuration file locally, remotely, or dynamically. Refer to the appropriate section, depending on the configuration option you selected when you started the Configuration Manager.

Saving a Configuration File in Local Configuration Mode

This section describes how to save a configuration created or modified in local mode to a file on the Site Manager workstation.

Note: If the Site Manager is in local configuration mode, and you want to store the configuration you have just created or modified to a file on the router, follow the instructions in this section to store the file locally. Then follow the instructions in the section *Transferring a Configuration File to the Wellfleet Router.*

To save your file, begin at the Wellfleet Configuration Manager Window, and complete the following steps.

1. Save the configuration file to the same directory in which the Site Manager is located, or to another directory.
   - — Select the File/Save option to save to the same directory (see Figure 18-1).
   - — Otherwise, select the File/Save As option and specify the appropriate directory.
2. Click on the OK button when the File Saved pop-up window appears.

Select the File/Exit option if you want to exit the Wellfleet Configuration Manager Window. The Configuration Manager prompts you to terminate this session; click the Ok button. You are returned to the Wellfleet Site Manager Window.

Refer to *Transferring a Configuration File to the Wellfleet Router* if necessary.

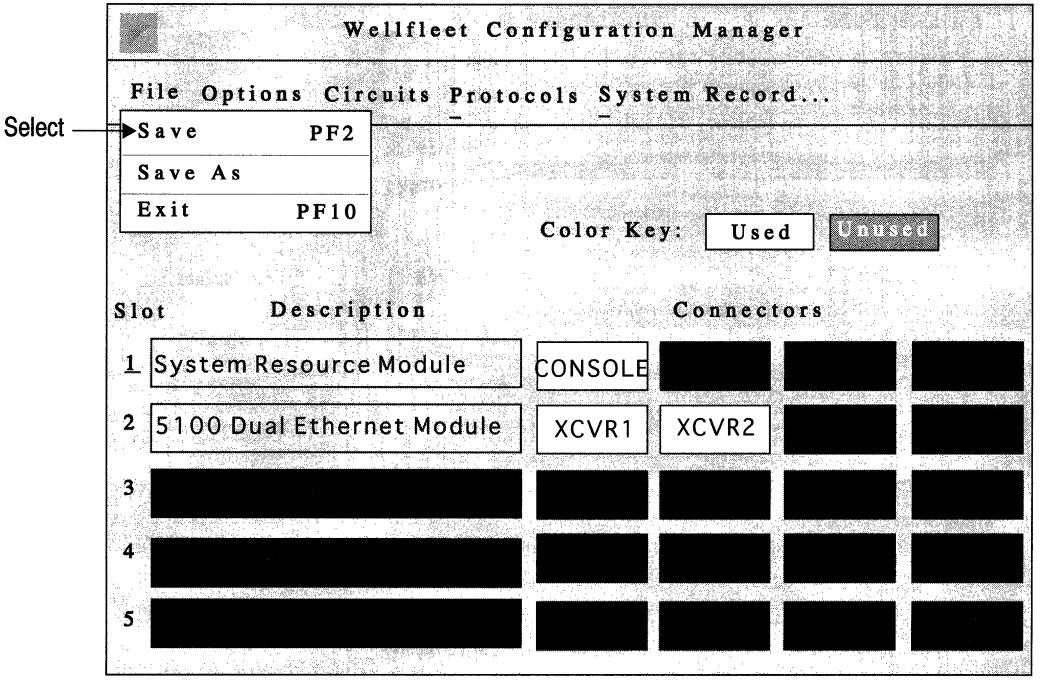


Figure 18-1. Saving the Configuration File

Saving a Configuration File in Remote Configuration Mode

This section describes how to save a configuration created or modified in remote configuration mode to a file on the router.

Warning: When you use the File/Save or File/Save As options in remote configuration mode, the configuration is automatically transferred to the router's file system. The destination system in a file transfer automatically overwrites any file already on its volume that has the same filename. If enough space does not exist on the file system for the new file, and the new file has the same name as an old file, the old file will be destroyed and the new file will be corrupt. This is because TFTP copies the new file over the old and runs out of space before completing the copy. Be sure to follow the instructions in this section to avoid corrupting the *config* file in the router if the Configuration Manager is in remote mode.

The recommended procedure for saving a *config* file created or modified in remote configuration mode is as follows:

1. If the file system on the router is an NVFS, compact the memory card to optimize the available space as follows:
   a. Select the Remote Files option in the Wellfleet Site Manager Window.
   b. Select Remote Commands/Compact.
   c. Click the OK button in the Confirmation Window.

   If the file system on the router is DOS, disregard this step.
2. Select the File/Save As option from the Wellfleet Configuration Manager Window.

   The Save Configuration File Window displays an Enter File name box.
3. Enter a new filename, such as *temp*, and click on the Save button. Do *not* use the filename *config*.

4. Click the Ok button when the File Saved pop-up window appears.

You replace an old configuration file with a new one as follows:

1. Verify the integrity of the new file first by booting with that file (refer to *Rebooting a Wellfleet Router with a Configuration File* later in this chapter).
2. Verify there is enough space on the volume for another copy by selecting the Remote Files option in the Wellfleet Site Manager Window.

   The Wellfleet File System Manager Window displays the files, file sizes, and available free space. The contiguous free space displayed in this window applies only to memory cards.
3. Ensure there is enough space on the volume for the file.

   DOS Instructions: Ensure the space occupied by new file is not larger than the available free space.

   NVFS Instructions: Ensure the space occupied by new file is not larger than the contiguous free space.
4. If enough space is available, copy the file to the old filename. Refer to the *Operations Guide: Site Manager* for detailed file management instructions.

Saving a Configuration File in Dynamic Configuration Mode

If you made changes to the currently active configuration file in the dynamic configuration mode, you may save these changes to a file on the router's file system. This preserves the current configuration file, yet gives you the option to reboot the router with these changes later. When you save these changes, the file is saved directly to the router.

To save your changes, start at the Wellfleet Configuration Manager Window, and complete the following steps:

1. Select the File/Save As option.

   The Save Configuration File Window displays an Enter File name box.

2. Enter a new filename to save the configuration file on the router, using the following format:

   filename.cfg

   filename is the name you are assigning this file, and **cfg** specifies the file type.

3. If the file system on the router is an NVFS, and the volume (slot location of the memory card on the router) that appears in the Volume box is not the volume to which you wish to save this file, click on the Volume box and select an alternate volume. Otherwise, go to the next step.

4. Click the Save button.

5. Click the Ok button when the File Saved pop-up window appears.

If you want to reboot the router with the file you just created and saved, refer to *Rebooting a Wellfleet Router with a Configuration File* for instructions.

Transferring a Configuration File to the Wellfleet Router

The Remote File System Manager allows you to transfer files between the Site Manager workstation and any Wellfleet router using TFTP. You must transfer a configuration file that has been configured and saved in local mode to the router before you can reboot the router with it.

Warning: The destination system in a file transfer automatically overwrites any file already on its volume that has the same filename. If enough space does not exist on the file system for the new file, and the new file has the same name as an old file, the old file will be destroyed and the new file will be corrupt. This is because TFTP copies the new file over the old and runs out of space before completing the copy. Be sure to follow the instructions in this section to avoid corrupting the *config* file.

You should choose a filename that is unique to the router when transferring the file. You will have an opportunity during the transfer procedure to specify a new filename. You can display the names of the files currently on the router by selecting the Remote Files option from the Wellfleet Site Manager Window.

The recommended procedure for transferring a file from the Site Manager workstation to a router is as follows:

1. If the file system on the router is an NVFS, compact the memory card to optimize the available space as follows:
    a. Select the Remote Files option in the Wellfleet Site Manager Window.
    b. Select Remote Commands/Compact.
    c. Click the OK button on the Confirmation Window.

    If the file system on the router is DOS, disregard this step.
2. Select the Remote Files option in the Wellfleet Site Manager Window.

The Wellfleet File System Manager Window appears.

3. Select the File/Tftp/Put File option.

   The (Local Filename) Tftp Put File Window appears (see Figure 18-2).

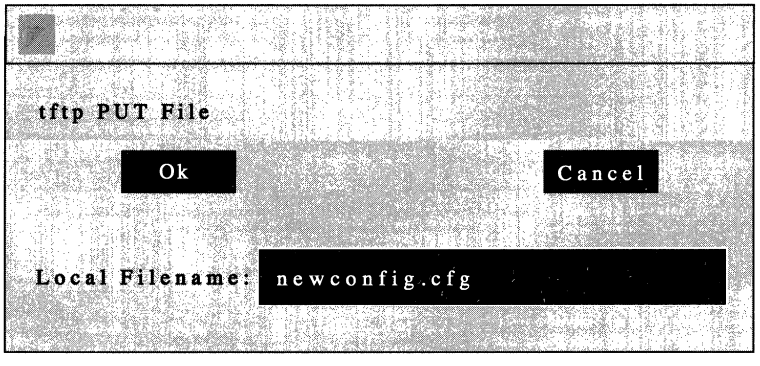


Figure 18-2. (Local Filename) Tftp Put File Window

4. Enter the name of the configuration file you wish to transfer to the router in the Local Filename box.
5. Click the Ok button.

   The (Remote Filename) Tftp Put File Window appears (see Figure 18-3).

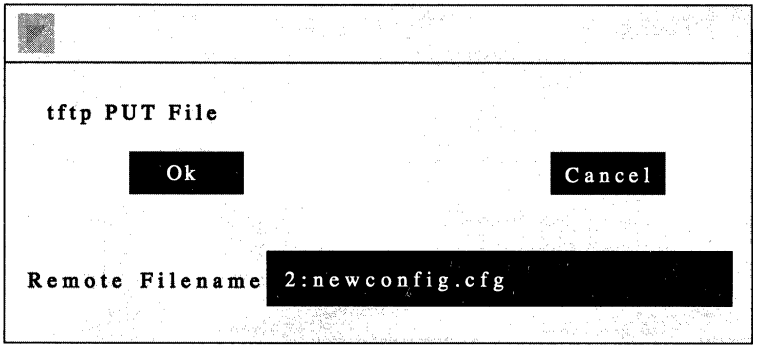


Figure 18-3. (Remote Filename) Tftp Put File Window

6. Use the following format to specify a volume and filename to store the configuration file on the router:

   <volume>:<filename.cfg>

   Where **<volume>** is *a* for a router with a DOS file system, or the slot location of the memory card to which you want to write, **<filename.cfg>** is the name you are assigning this file for storage on the router, and the file type.

   Note: Wellfleet strongly recommends that you do not transfer a file to a router's file system that has the same filename as an existing file. The TFTP Put File (Remote Filename) Window allows you to change the destination filename.

7. Click the Ok button.
8. Click the Ok button again when the confirmation window appears.

   The file is now transferred from the Site Manager workstation to the appropriate volume on the router. When this process is complete, a pop-up window indicates that the command completed successfully.

9. Click the Ok button.

Now that you have transferred your configuration file to the router, you can reboot the router with it. The next section provides instructions.

You replace an old configuration file with a new one as follows:

1. Verify the integrity of the new file first by booting with that file.
2. Verify there is enough space on the volume for another copy by selecting the Remote Files option from the Wellfleet Site Manager Window.

   The Wellfleet File System Manager Window displays the files, file sizes, and available free space. The contiguous free space displayed in this window applies only to memory cards.
3. Ensure there is enough space on the volume for the file.

   DOS Instructions: Ensure the space occupied by new file is not larger than the available free space.

   NVFS Instructions: Ensure the space occupied by new file is not larger than the contiguous free space.
4. If enough space is available, copy the file to the old filename. Refer to the *Operations Guide: Site Manager* for detailed file management instructions.

Rebooting a Wellfleet Router with a Configuration File

After you save a configuration file to the router, you implement the configuration by rebooting the router with that configuration file. You begin from the Wellfleet Site Manager Window and proceed as follows:

1. Select the Admin/Boot option.

   The Boot Window appears (see Figure 18-4). The default volume is displayed next to the default boot image file *(boot.exe* or *ace.out)* and default configuration file (*config*). The default volume is the first available memory access card (indicated by slot number) on a router with an NVFS or *a* on a router with a DOS file system.

   Refer to step 2 if you want to boot from the default volumes and configuration file.

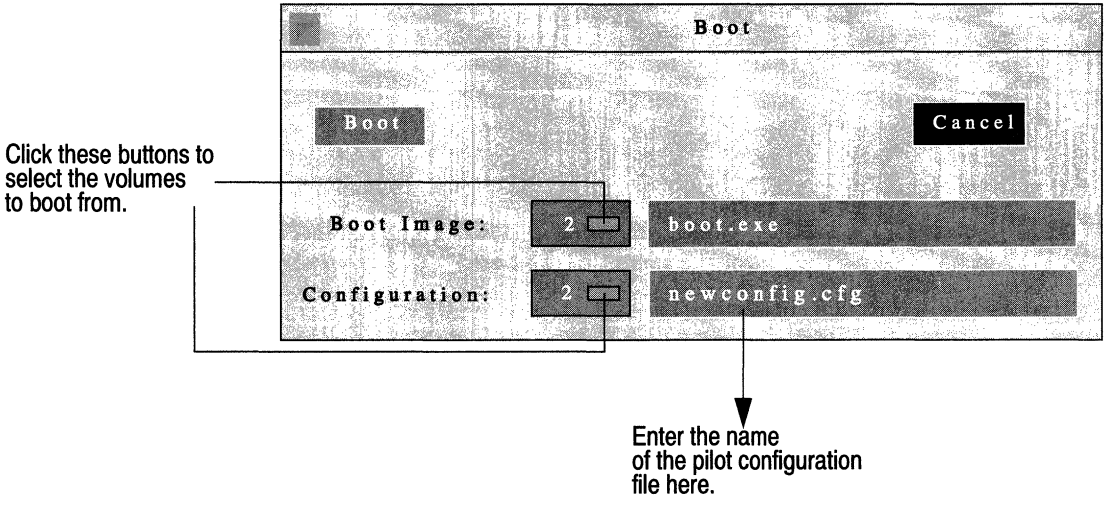


Figure 18-4. Boot Window

If the router has multiple volumes, you can select one volume from which to read the boot image and another from which to read the configuration file as follows:

a. Click the rectangle adjacent to the Boot Image volume number.

   A pop-up window displays the available volumes containing the boot image.

b. Click the number of the slot you want to boot from.

   The pop-up window closes and the new slot number is displayed next to Boot Image.

c. Click the rectangle adjacent to the Configuration slot number.

   A pop-up window displays the available slots containing the configuration file.

d. Click the number of the slot you want to configure from.

   The pop-up window closes and the new slot number is displayed.

Enter the configuration filename in the Configuration filename box if you want to configure from an alternative file. When you select the Boot button, the router boots and configures with the image and configuration file in the volumes displayed.

2. Click on the Boot button.

The router boots using the software image and the configuration file you specified.

Note: The software image and configuration file revert to their respective default volumes and filenames (*boot.exe* or *ace.out* and *config*) after every boot. To change the default boot or configuration file, back up the old default file using the copy option; then overwrite the old default file with the new default file using the copy option.

After you successfully reboot the router with a configuration file, and it is up and running on your network, it is actively routing and/or bridging traffic.

Appendix A

Site Manager Default Settings

Site Manager Default Settings

About this Appendix

This appendix lists the Wellfleet-provided default settings for the Site Manager. Use the Configuration Manager to edit any of the Site Manager default settings listed here.

Circuit Parameters

Software Version 7.50 supports Ethernet, FDDI, synchronous, E1, T1, Token Ring, and HSSI circuits. The following sections describe the Site Manager default parameter settings for these circuits.

Ethernet Circuit Parameters

Parameter	Default Setting
Enable	Enable
BOFL Enable	Enable
BOFL Timeout	5

FDDI Circuit Parameters

Parameter	Default Setting
Enable	Enable
BOFL Enable	Enable
BOFL Timeout	5
SMT Connection Policy	65381
SMT Notify	22 (seconds)
MAC TReq	2062500 (octet units)

Synchronous Circuit Parameters

Parameter	Default Setting
Enable	Enable
BOFL	Enable
BOFL Timeout	5
MTU	1600
Promiscuous	Disable
Clock Source	External
Clock Speed	64K
Signal Mode	Balanced
RTS Enable	Disable
Burst Count	Enable
Service	LLC1
Minimum Frame Spacing	1
Local Address	Explicit
Pass Thru Local Mac Address	None
Remote Address	Explicit
Pass Thru Remote Mac Addr	None
WAN Protocol	Standard
CRC Size	16
Sync Media	1

E1 Circuit Parameters

Parameter	Default Setting
Enable	Enable
HDB3S Support	Disable
Clock Mode	Internal
Mini Dacs	Idle

T1 Circuit Parameters

Parameter	Default Setting
Enable	Enable
Frame Type	ESF
B8ZS Support	Disable
Line Buildout	1 (foot)
Clock Mode	Internal
Mini Dacs	Idle

Token Ring Circuit Parameters

Parameter	Default Setting
Enable	Enable
MAC Address Override	None
MAC Address Select	Boxwide
Speed	16M Bps

HSSI Circuit Parameters

Parameter	Default Setting
Enable	Enable
BOFL	Enable
BOFL Frequency	1 (second)
MTU (Maximum Transfer Unit)	4495 bytes
Transmission Interface	DS3
External Clock Speed	46359642 (44.736Mbps)
CRC Size	32-bit

Frame Relay Parameters

This section describes the Site Manager default interface and permanent virtual circuit parameter settings for Frame Relay.

Frame Relay Interface Parameters

Parameter	Default Setting
Enable	Enable
Mgmnt Type	ANSI T1 617D
Address	Addr Q922
Address Length	Two Byte
Polling Interval	10 (seconds)
Full Enquiry Interval	6
Error Threshold	3
Monitored Events	4
Multicast	Disable

Frame Relay Permanent Virtual Circuit Parameters

Parameter	Default
Circuit State	Invalid
Multicast	Unicast
Mode	Group Access

SMDS Parameters

This section describes the Site Manager default interface parameter settings for SMDS.

SMDS Interface Parameters

Parameter	Default Setting
Enable	Enable
Individual Address	None
Group Address	None
ARP Address	None
Heartbeat Poll	Disable
Heartbeat Poll Interval	10 (seconds)
Heartbeat Poll Downcount	3
LMI Network Mgmt	Disable

AppleTalk Parameters

This section describes the Site Manager default global and interface parameter settings for AppleTalk.

AppleTalk Global Parameters

Parameter	Default Setting
Enable	Enable

AppleTalk Interface Parameters

Parameter	Default Setting
Port Enable	Yes
Checksum Enable	No
TR End Station	No
Node ID	253
Network ID	0
Network Start	0
Network End	0
Default Zone	None
Zone List	None

Bridge Parameters

This section describes the Site Manager default global and interface parameter settings for the Transparent/Translating Bridge (including Spanning Tree).

Filtering is an optional feature. See *Configuring Filters* for instructions on creating Bridge filters.

Bridge Global Parameters

Parameter	Default Setting
Enable	Enabled if at least one interface is enabled with the Bridge

Bridge Interface Parameters

Parameter	Default Setting
Enable	Enable

Spanning Tree Global Parameters

Parameter	Default Setting
Spanning Tree Global Enable	Enable
Bridge Priority	None
Bridge MAC Address	None
Max Age	2000
Hello Time	200
Forward Delay	1500

Spanning Tree Interface Parameters

Parameter	Default Setting
Spanning Tree Interface Enable	Enable
Priority	128
Path Cost	1

Source Routing Parameters

This section describes the Site Manager default global and interface parameter settings for Source Routing.

Filtering is an optional feature. See *Configuring Filters* for instructions on creating Source Routing filters.

Source Routing Global Parameters

Parameter	Default Setting
Enable	Enabled if at least one interface is enabled with the Bridge
SR Bridge Internal LAN ID	None
SR Bridge ID	None
SR Group LAN ID	4095
IP Encapsulation	Disable
Conn. IP NTWK Ring Number	None
IP Net Mtu	4562

Source Routing Interface Parameters

Parameter	Default Setting
Enable	Enable
Max Number of RDs	7
Source Routing Ring Number	0
Outbound STEs	Accept
Inbound STEs	Accept
Frames with IP Ring	Accept

DECnet Phase IV Router Parameters

This section describes the Site Manager default global and interface parameter settings for the DECnet Phase IV router.

Static Adjacencies and Filtering are optional features. See *Configuring DECnet Phase IV* for instructions on configuring static adjacencies. See *Configuring Filters* for instructions on configuring DECnet filters.

DECnet Phase IV Global Parameters

Parameter	Default Setting
Route Enable	Enable
Broadcast Route Timer	180
Route Max Addr	1023
Max Broadcast NonRouters	64
Max Broadcast Routers	32
Max Circuits	1024
Max Cost	1022
Max Hops	30
Max Visits	63
Area Max Cost	1022
Area Max Hops	30
Max Area	63

DECnet Phase IV Interface Parameters

Parameter	Default Setting
Enable	Enable
Area ID	None
Node ID	None
Circuit Cost	10
Hello Timer	15
Max Routers	33
Router Priority	64
End Nodes MAC	None
End Routers MAC	None
Area Routers MAC	None
Node Hello	Enable
Router Hello	Enable
Topology Update	Enable

IP Parameters

This section describes the Site Manager default global and interface parameter settings for IP. The IP router is enabled to support ARP and TFTP.

RIP, Adjacent Hosts, Static Routing, route filters, and traffic filters are optional features. See *Configuring IP* for instructions on enabling the first four features; see *Configuring Filters* for instructions on configuring IP filters.

IP Global Parameters

Parameter	Default Setting
Enable	Set to Enable when you add IP to a circuit.
Forwarding	Forwarding
ARP Forwarding	Forwarding
Default TTL	30
RIP Diameter	15
Route Cache Flush Interval	1 Minute
MIB Table(s) Maintained	Routing

IP Interface Parameters

Parameter	Default Setting
Enable	Enable
Subnet Mask	None
Broadcast Address	None
Interface Cost	1
MTU Discovery	Off
Address Mask Reply	On
All Subnet Bcast	On
Address Resolution	Enable
Proxy	Off
Host Cache	1 (off)
Checksum	On
MAC Address	None
TR End Station	Disable
SMDS Group Address	None
SMDS Arp Req Address	None
FR Broadcast DLCI	0
FR Multicast DLCI #1	0
FR Multicast DLCI #2	0
Redirects	Enable
Encapsulation	Ethernet

RIP Interface Parameters

Parameter	Default Setting
Enable	Enable
RIP Supply	Enable
RIP Listen	Enable
Default Route Supply	Disable
Default Route Listen	Disable
Poisoned Reverse	Poisoned

TFTP Parameters

Parameter	Default Setting
Enable	Enable
Default Volume	2
Retry Time Out	5
Close Time Out	25
Retransmit	5

OSPF Parameters

This section describes the Site Manager default global, area, interface, and virtual link parameter settings for OSPF.

Area Ranges is an optional feature of OSPF. For information on configuring area ranges, see *Configuring OSPF.*

OSPF Global Parameters

Parameter	Default Setting
Enable	Enable
Router ID	None
AS Boundary Router	No
Hold Down Timer	1
OSPF Slot	All slots

OSPF Area Parameters

Parameter	Default Setting
Enable	Enable
Authentication Type	None
Import AS Extern	Yes
Stub Metric	1
Import Summaries	True

OSPF Interface Parameters

Parameter	Default Setting
Enable	Enable
Area ID	0.0.0.0
Type	Broadcast
Router Priority	1
Transit Delay	1 (second)
Retransmit Interval	5 (seconds)
Hello Interval	10 (seconds)
Dead Interval	40 (seconds)
Poll Interval	120 (seconds)
Metric Cost	1
Password	None

OSPF Virtual Link Parameters

Parameter	Default Setting
Enable	Enable
Transit Delay	1
Retransmit Interval	5
Hello Interval	15
Dead Interval	60
Password	None

IPX Parameters

This section describes the Site Manager default global and interface parameter settings for IPX.

RIP, Adjacent Hosts, Static Routing, network level and server level SAP filters, and traffic filters are optional features. See *Configuring IPX* for instructions on enabling the first five features; see *Configuring Filters* for instructions on configuring IPX filters.

IPX Global Parameters

Parameter	Default Setting
Enable	Enable
Host Number	automatically generated, unique 6-byte host number based on Wellfleet Router's serial number. It is not displayed.

IPX Interface Parameters

Parameter	Default Setting
Enable	Enable
Cost	1 for each hop
Xsum On	On
Cfg Encaps	On
TR End Station	Enable
NetBIOS Accept	Enable
NetBIOS Deliver	Enable
WAN SAP Period	1
FR Broadcast	ffffff (not displayed)
FR Multicast	ffffff (not displayed)
Split Horizon	Enable

IPX RIP Interface Parameters

Parameter	Default Setting
Enable	Enable
Supply	Enable
Listen	Enable

SNMP Parameters

This section describes the Site Manager default global parameter settings for SNMP.

SNMP Global Parameters

Parameter	Default Setting
Enable	Enable
Use Lock	Enable
Lock Timeout	2
Authentication Failure Trap	Enable
Trap Debug events	Off
Trap Trace events	Off
Trap Info events	On
Trap Warning events	On
Trap Fault events	On

SNMP Community Parameters

During initialization, if the SNMP agent detects no valid community with at least one manager, the agent automatically configures a read-write public community with a wild card manager (0.0.0.0) to ensure that the Wellfleet Router is always SNMP manageable. For security reasons, Wellfleet recommends that you replace the public community and wild card manager with a unique community configured with a limited list of managers. See *Configuring the SNMP Agent* for instructions on how to configure SNMP communities and managers.

VINES Parameters

This section describes the Site Manager default global and interface parameter settings for VINES.

Filtering is an optional feature. See *Configuring Filters* for instructions on configuring VINES filters.

VINES Global Parameters

Parameter	Default Setting
Enable	Enable
Network ID	None
Broadcast Class	All

VINES Interface Parameters

Parameter	Default Setting
Enable	Enable
Interface Type	Ethernet
ARP Enable	Disable
End Station Enable	Disable
Ethernet Header	Ethernet
Remote Client Enable	Disable

XNS Parameters

This section describes the Site Manager default global and interface parameter settings for XNS.

RIP, Adjacent Hosts, Static Routing, and traffic filters are optional features. See *Configuring XNS* for instructions on enabling the first three features; see *Configuring Filters* for instructions on configuring XNS filters.

XNS Global Parameters

Parameter	Default Setting
Enable	Enable
Host Number	Base host number you entered when first adding XNS to the circuit

XNS Interface Parameters

Parameter	Default Setting
Enable	Enable
Cost	1 for each hop
Xsum On	Enable
MAC Address	None
SMDS Group Address	None
Ext Server	Disable
Ex Serv Network	Enable
Ex Serv Host ID	0
Ex Serv Pkt Type	None
Ex Serv SockNM	None
Frame Relay Broadcast	ffffff (not displayed)
Frame Relay Multicast	ffffff (not displayed)

XNS RIP Interface Parameters

Parameter	Default Setting
Enable	Enable
Supply	Enable
Listen	Enable

Protocol Prioritization Parameters

This section describes the Site Manager default content-based priority filter, length-based priority filter, and interface parameter settings for Protocol Prioritization.

Protocol Prioritization Configuration Parameters

Parameter	Default Setting
Content-Based Priority	Yes
Length-Based Priority	Yes

Length-Based Priority Filter Parameters

Parameter	Default Setting
Enable	Enable
Packet Length	Length value supplied when this filter was created
Less Than or Equal Queue	Normal
Greater Than Queue	Low

Protocol Prioritization Interface Parameters

Parameter	Default Setting
Enable	Enable
High Queue	20 (packets)
Normal Queue	20 (packets)
Low Queue	20 (packets)
Max High Queue Latency	250 (ms)

Technician Interface Console Parameters

The Site Manager default settings for the Technician Interface console are as follows:

Parameter	Default Setting
Baud Rate	9600
Data Bits	8
Parity	None
Stop Bits	1
Enable Modem	Disable
Lines Per Screen	24
Enable MORE	Enable
Prompt	ti>
Login Timeout	1 minute
Password Timeout	1 minute
Command Timeout	15 minute
Login Retries	3 times

Appendix B

IEEE Assigned Codes

List of Tables

IEEE Assigned Codes

About This Appendix

This appendix provides you with several tables of various IEEE assigned codes: protocol/packet type assignments, publicly listed vendor codes and sample service access points. Please note that complete validity is not guaranteed.

Protocol/Packet Type Assignments

Table B-1 provides a list of packet types and the protocol/company to which they are assigned.

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
0000 - 05DC	IEEE 802.3 Length Field
0101 - 01FF	Experimental
0200	Xerox PUP
0201	Xerox PUP Address Translation
0400	Nixford
0600	Xerox XNS IDP
0800	DOD Internet Protocol (IP)

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
0801	X.25 Internet
0802	NBS Internet
0803	ECMA Internet
0804	CHAOSnet
0805	X.25 Level 3
0806	Address Resolution Protocol (for IP and CHAOSnet)
0807	XNS Compatibility
081C	Symbolics
0888 - 088A	Xyplex
0900	Ungerman-Bass Network Debugger
0A00	Xerox IEEE 802.3 PUP
0A01	Xerox IEEE 802.3 PUP Address Translation
0BAD	Banyan VINES IP
0BAE	Banyan VINES Loopback
0BAF	Banyan VINES Echo
1000	Berkeley Trailer Negotiation
1001 - 100F	Berkeley Trailer Encapsulation for IP
1600	VALID System Protocol
4242	PCS Basic Block Protocol
5208	BBN SIMnet
6000	DEC experimental
6001	DEC Maintenance Operation Protocol (MOP) Dump/Load Assistance

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
6002	DEC MOP Remote Control
6003	DEC DECnet Phase IV
6004	DEC Local Area Transport (LAT)
6005	DEC DECnet Diagnostics
6006	DEC DECnet Customer Use
6007	DEC Local Area VAX Cluster (LAVC)
6008	DEC AMBER
6009	DEC MUMPS
6010 - 6014	3COM Corporation
7000	Ungerman-Bass Download
7001	Ungerman-Bass Diagnostic/Loopback
7002	Ungerman-Bass Diagnostic/Loopback
7020 - 7029	LRT
7030	Proteon
7034	Cabletron
8003	Cronus VLN
8004	Cronus Direst
8005	Hewlett Packard Probe
8006	Nestar
8008	AT&T
8010	Excelan
8013 - 8016	Silicon Graphics
8019	Apollo Domain

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
802E	Tymshare
802F	Tigan, Inc.
8035	Reverse Address Resolution Protocol (RARP)
8036	Aeonic Systems
8038	DEC Spanning Tree - RBMS
8039	DEC DSM/DTP
803A	DEC Argonaut Console
803B	VAXELN
803C	DEC DNA Naming Service
803D	CSMA/CD Encryption Protocol
803E	DEC DNA Time Service
803F	DEC LAN Traffic Monitor Protocol
8040	DEC NetBIOS Emulator
8041	DEC Local Area System Transport
8042	DEC for future use
8044	Planning Research Corp.
8046 - 8047	AT&T
8049	ExperData
805B	Versatile Message Translation Protocol RFC 1045
805C	Stanford V Kernel, production
805D	Evans and Sutherland
8060	Little Machines
8062	Counterpoint Computers

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
8065 - 8066	University of Massachusetts at Amhearst
8067	Veeco Integrated Automation
8068	General Dynamics
8069	AT&T
806A	Autophon
806C	ComDesign
806D	Compugraphic Corporation
806E - 8077	Landmark Graphics Corp.
807A	Matra
807B	Dansk Data Elektronic
807C	Merit Internodal
807D - 8080	Vitalink Communications Bridge Management
8081 - 8083	Counterpoint Computers
809B	Kinetics Ether Talk (Apple Talk over Ethernet)
809C - 809E	Datability
809F	Spider Systems Ltd.
80A3	Nixdorf Computers
80A4 - 80B3	Siemans Gammasonics Inc.
80C0 - 80C3	Digital Communications Assoc. Inc.
80C4	Banyan VINES IP
80C5	Banyan VINES Echo
80C6	Pacer Software
80C7	Applitek

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
80C8 - 80CC	Integraph Corporation
80CD -80CE	Harris Corporation
80CF - 80D2	Taylor Instrument
80D3 - 80D4	Rosemount Corporation
80D5	Ungerman-Bass
80DD	Varian Associates
80DE	Integrated Solutions Transparent Remote File System (TRFS)
80DF	Integrated Solutions
80E0 - 80E3	Allen-Bradley
80E4 - 80F0	Datability
80F2	Retix
80F3	Kinetics Apple Talk Address Resolution Protocol (AARP)
80F4 - 80F5	Kinetics
80F7	Apollo Computer
80FF - 8103	Wellfleet Communications
8107 - 8109	Symbolics
8130	Waterloo Microsystems Inc.
8131	VG Laboratory Systems
8137 - 8138	Novell, Inc.
8139 - 813D	KTI
814C	SNMP
9000	DEC MOP LAN Loopback

Table B-1. Protocol/Packet Type Assignments

Packet Type	Description
9001	Bridge Communications XNS Systems Management
9002	Bridge Communications TCP/IP Systems Management
9003	Bridge Communications loop detection
FF00	BBN VITAL LANBridge cache wakeups

Publicly Listed Vendor Codes

All vendors are required to register their equipment with the IEEE 802 committee for assignment of ethernet address blocks (vendor codes). Table B-2 provides a list of the publicly listed vendor codes. Note that vendors are not required to publicly list these codes.

Table B-2. Publicly Listed Vendor Codes

Code	Vendor
00000C	Cisco
00000F	NeXT
000010	Sytek
00001D	Cabletron
000020	DIAB (Data Industrier AB)
000022	Visual Technology
00002A	TRW
00003D	AT&T
000055	AT&T
00005A	S & Koch
00005E	IANA
000065	Network General
00006B	MIPS
000077	MIPS
00007A	Ardent
000089	Cayman Systems (Gatorbox)
000093	Proteon
00009F	Ameristar Technology

Table B-2. Publicly Listed Vendor Codes

Code	Vendor
0000A2	Wellfleet
0000A3	Network Application Technology
0000A6	Network General (internal assignment, not for products)
0000A7	NCD (X-terminals)
0000A9	Network Systems
0000AA	Xerox (Xerox machines)
0000B3	CIMlink
0000B7	Dove (Fastnet)
0000BC	Allen-Bradley
0000C0	Western Digital
0000C6	HP Intelligent Networks Operation (formerly EON Systems)
0000C8	Altos
0000C9	Emulex (Terminal Servers)
0000D7	Dartmouth College (NED Router)
0000DD	Gould
0000DE	Unigraph
0000E2	Acer Counterpoint
0000EF	Alantec
0000FD	High Level Hardware (Orion, UK)
000102	BBN (BBN internal usage - not registered)
001700	Kabel
00802D	Xylogics, Inc. (Annex Terminal Servers)

Table B-2. Publicly Listed Vendor Codes

Code	Vendor
00808C	Frontier Software Development
0080C2	IEEE 802.1 Committee
0080D3	Shiva
00AA00	Intel
00DD00	Ungermann-Bass
00DD01	Ungermann-Bass
020701	Racal InterLan
020406	BBN (BBN internal usage - not registered)
026086	Satelcom MegaPac (UK)
02608C	3Com (IBM PC; Imagen; Valid; Cisco)
02CF1F	CMC (Masscomp; Silicon Graphics; Prime EXL)
080002	3 Com (formerly Bridge)
080003	ACC (Advanced Computer Communications)
080005	Symbolics (Symbolics LISP Machines)
080008	BBN
080009	Hewlett-Packard
08000A	Nestar Systems
08000B	Unisys
080010	AT&T
080011	Tektronix, Inc.
080014	Excelan (BBN Butterfly; Masscomp; Silicon Graphics)
080017	NCS
08001A	Data General

Table B-2. Publicly Listed Vendor Codes

Code	Vendor
08001B	Data General
08001E	Apollo
080020	SUN (SUN Machines)
080022	NBI
080025	CDC
080026	Norsk Data (Nord)
080027	PCS Computer Systems GmbH
080028	TI (Explorer)
08002B	DEC
08002E	Metaphor
08002F	Prime Computer (Prime 50-Series LHC300)
080036	Integraph (CAE stations)
080037	Fujitsu-Xerox
080038	Bull
080039	Spider Systems
080041	DCA Digital Comm. Assoc.
080046	Sony
080047	Sequent
080049	Univation
08004C	Encore
08004E	BICC
080056	Stanford University
08005A	IBM

Table B-2. Publicly Listed Vendor Codes

Code	Vendor
080067	Comdesign
080068	Ridge
080069	Silicon Graphics
08006E	Excelan
080075	DDE (Danish Data Elektronik A/S)
08007C	Vitalink (TransLAN III)
080080	XIOS
080086	Imagen/QMS
080087	Xyplex (Terminal Servers)
080089	Kinetics (AppleTalk Ethernet interface)
08008B	Pyramid
08008D	XYVision (XyVision Machines)
080090	Retix, Inc. (Bridges)
800010	AT&T
AA0003	DEC (Global physical address for some DEC machines)
AA0004	DEC (Local logical address for systems running DECnet)

Sample Service Access Points

Table B-3 provides a list of sample service access points.

Table B-3. Sample Service Access Points

SAP Value	Usage
00	Null LSAP
FF	Globe SAP
02	Individual LLC sublayer management function
03	Group LLC sublayer management
FE	OSI network layer protocol
80	3COM protocol (XNS)
04	SNA protocol
05	SNA path control group SAP
E0	Novell NetWare
F0	IBM NetBIOS protocol
F8/FC	IBM remote initial program load (IPL)
AA	TCP/IP SNAP protocol
06	Arpanet's internet protocol
14	PROWAY-LAN
78	EIA-RS 511
94	ISI IP
142	PROWAY-LAN
170	SNAP
254	ISO CLNS IS 8473
255	Global DSAP

These numbers have partially been obtained by RFC 1340. These number (and others) are assigned by the IEEE Standards Office. The address is: IEEE Standards Office, 345 East 47th Street, New York, NY 10017, Attn. Vince Condello. The phone number is: (212) 705-7092.

Appendix C

Converting Version 5 Traffic Filters

List of Figures

Converting Existing Traffic Filters

About this Appendix

This appendix is provided for those Wellfleet users who have in the past configured their traffic filters using the Version 5 platform Configuration Editor tool. It aims to provide a way to easily and logically create filters in Version 7 that are equivalent to the existing Version 5 traffic filters. Use this appendix if you want to:

- Understand the differences between the Version 5 and Version 7 traffic filter schemes
- Understand the benefit of using the Version 7 traffic filter scheme
- Use the *manual* conversion algorithm to convert your existing Version 5 traffic filters to Version 7 traffic filters

Traffic Filter Scheme Differences

There are two things in the Version 5 traffic filters scheme that the Version 7 traffic filters scheme does not support; they are:

- The Accept action

  When configuring traffic filters in the Version 5 platform, you can specifically configure filters for the traffic that you want to be accepted. To do this, you configure a filter with the Accept action. This is normally done when a limited amount of traffic is to be accepted, while all other traffic is to be dropped.

  In the Version 7 platform, there is no Accept action. This may at first seem confusing; however, the key to configuring traffic filters in the Version 7 platform is to think in terms of what types of traffic you want simply to be accepted; then configure the appropriate filters for all *other* types of traffic.

- Precedence

  The second difference between the Version 5 and Version 7 traffic filters schemes is that with Version 7, you can no longer assign a precedence to filters on the same circuit or network interface. In Version 7, when there are multiple filters on a circuit, all filters are taken into account. *It is up to the user not to configure conflicting filters on the same circuit or network interface.*

These two differences do not affect Version 7 traffic filter capability; you can configure any filter in the Version 7 platform that you could in the Version 5 platform. It is a matter of thinking about your traffic filter configuration process differently.

In the Version 7 platform, you configure filters for everything you want dropped, logged, dropped and logged, or for any traffic that requires special instruction (for example, forwarding to the next hop, or forwarding to a circuit list). Everything else is accepted and forwarded as it normally would be, as if there were no filters on the circuit. This means that you do not have to configure a filter for traffic that you wish to accept, as you did in the Version 5 platform.

To facilitate the configuration of Version 7 traffic filters that are identical to your existing Version 5 traffic filters, Wellfleet provides a *manual* conversion algorithm (discussed in a subsequent section).

Benefit of Using the Version 7 Traffic Filter Scheme

Aside from being able to configure any traffic filter that you can configure with the Version 5 platform, the Version 7 provides an additional benefit. The benefit is that in the presence of filters, the Version 7 scheme produces a higher forwarding rate compared to the Version 5 scheme. The Version 5 scheme imposes an additional per-packet processing overhead because *all* packets match a filter. In the Version 7 scheme, traffic that is simply accepted requires no filter; therefore, those packets do not incur this overhead.

Creating Version 7 Filters

If you need to create Version 7 traffic filters that are identical to your existing Version 5 traffic filters, there are two approaches.

- You can first identify the fields/ranges of the traffic that you want to drop, log, drop and log, or for which you wish to assign special instruction (for example, forwarding to the next hop, or forwarding to a circuit list). With that information, you can use the Configuration Manager to create the necessary filters. This is described in *Configuring Filters*.

- You can use the *manual* conversion algorithm provided in the next section to determine what fields/ranges in your Version 5 traffic filters require you to create a corresponding Version 7 traffic filter. Then, with that information, you can follow the instructions in *Configuring Filters* to create the necessary filters.

Note: Following the conversion algorithm as specified guarantees completeness in the filter conversion process. It is, however, an exhaustive algorithm, and you may not find it necessary to go through the algorithm to configure the appropriate filters for your network. Wellfleet suggests that you try the first approach, and use the algorithm only if you cannot easily identify the fields/ranges of traffic that you want to drop, log, or assign special instruction.

The Conversion Algorithm

The conversion algorithm allows you to take your currently configured Version 5 filters and identify which of them require a filter to be created in the Version 7 scheme to achieve identical results. The Version 5 traffic filters in Figure C-1 will be considered in the following explanation of the conversion algorithm steps.

Filter Rule 1	Filter Rule 2	Filter Rule 2
Precedence:3 Action: Accept Fields/Ranges: IP Dest: 192.32.32.32 Protocol: UDP (17) UDP Port: SUNRPC (111)	Precedence:3 Action: Accept Fields/Ranges: Protocol: TCP (6) UDP Port: FTP (21)	Precedence:1 Action: Drop

Summary

Accept SUNRPC traffic for 192.32.32.32.
Accept FTP traffic from any station.
Drop everything else.

Performance Implications

Every dropped packet matches 1 rules.
Every accepted packet matches 2 rules.
When a rule matches, it's action cannot be taken until it is determined that it is the highest precedence.
Determining which filter(s) triggered and what action to take adds processing overload for each packet.

Figure C-1. Example of Version 5 Filters

1. Enumerate all fields used within all Version 5 traffic filters for a given interface.

   For the Version 5 filters in Figure C-1, the fields are:

   — IP Destination

   — Protocol

   — Port

2. Enumerate all ranges for these fields across all filters.

   For the Version 5 filters in Figure C-1, the ranges would be:

   — IP Destination:

   0 -> 192.32.32.31

   192.32.32.32 -> 192.32.32.32

   192.32.32.33 -> 255.255.255.255

   — Protocol

   0 -> (TCP - 1)

   TCP -> TCP

   (TCP +1) -> (UDP -1)

   UDP -> UDP

   (UDP + 1) -> 0xff

   — Port

   0 -> (FTP -1)

   FTP -> FTP

   (FTP + 1) -> (SUNRPC -1)

   SUNRPC -> SUNRPC

   (SUNRPC + 1) -> 0XFFFF

3. Create a matrix of each possible combination of fields and ranges.

   Figure C-2 shows a partial field/range matrix for the fields and ranges enumerated in step 2. Note that the complete matrix contains 75 entries.

	IP Destination value	Protocol value	Port value
Combination 1 =	0 -> 192.32.1.253	0 -> (TCP - 1)	0 -> (FTP 1)
Combination 2 =	0 -> 192.32.1.253	0 -> (TCP - 1)	FTP ->FTP
Combination 3 =	0 -> 192.32.1.253	0 -> (TCP - 1)	(FTP + 1) -> (SUNRPC -1)
Combination 4 =	0 -> 192.32.1.253	0 -> (TCP - 1)	SUNRPC -> SUNRPC
Combination 5 =	0 -> 192.32.1.253	0 -> (TCP - 1)	(SUNRPC + 1) -> 0XFFFF
Combination 6 =	0 -> 192.32.1.253	TCP -> TCP	0 -> (FTP 1)
Combination 7 =	0 -> 192.32.1.253	TCP -> TCP	FTP ->FTP
Combination 8 =	0 -> 192.32.1.253	TCP -> TCP	(FTP + 1) -> (SUNRPC -1)
Combination 9 =	0 -> 192.32.1.253	TCP -> TCP	SUNRPC -> SUNRPC
Combination 10 =	0 -> 192.32.1.253	TCP -> TCP	(SUNRPC + 1) -> 0XFFFF

Figure C-2. Field/Range Matrix

4. Cycle through and compare each combination to the existing Version 5 traffic filters. Then, for each of these combinations, refer to the following list and take the appropriate action.

— If no Version 5 filter rule matches the combination, ignore it. No Version 7 traffic filter is necessary.

— If the highest precedence Version 5 filter rule that matches the combination has the action *Accept*, ignore it. No Version 7 traffic filter is necessary.

— If the highest precedence Version 5 filter rule that matches the combination has an action *other than Accept*, a Version 7 traffic filter must be configured with the corresponding action.

  For example, if a combination matches a Version 5 filter rule that has the action Drop and Log, you must create a Version 7 traffic filter for this combination with the action Drop and Log.

  Check to see if this filter rule can be aggregated with other filter rules that have the *exact* same action. If it can, then do so. For example, combinations 1 through 6 in Figure C-2 all resolve to the Drop action; therefore, they should be aggregated as shown in Figure C-3.

Fields	IP Destination: 0 -> 192.32.32.31 Protocol: 0 -> TCP
Action	Drop

Figure C-3. Example of Filter Aggregation

  Because the Port value could be anything in this example, it is removed from the definition. If this exercise were continued, this filter rule would be aggregated again with other combinations.

— If a combination matches a Version 5 filter rule with an action of Accept and Log, ignore the Accept part, but configure a Version 7 traffic filter with the action Log.

Note: Remember that in the Version 7 platform, it is up to the user to avoid configuring conflicting rules and actions on the same circuit.

In this example, the Version 7 traffic filters shown in Figure C-4 are the filters that result from using the conversion algorithm on the Version 5 traffic filters shown in Figure C-1.

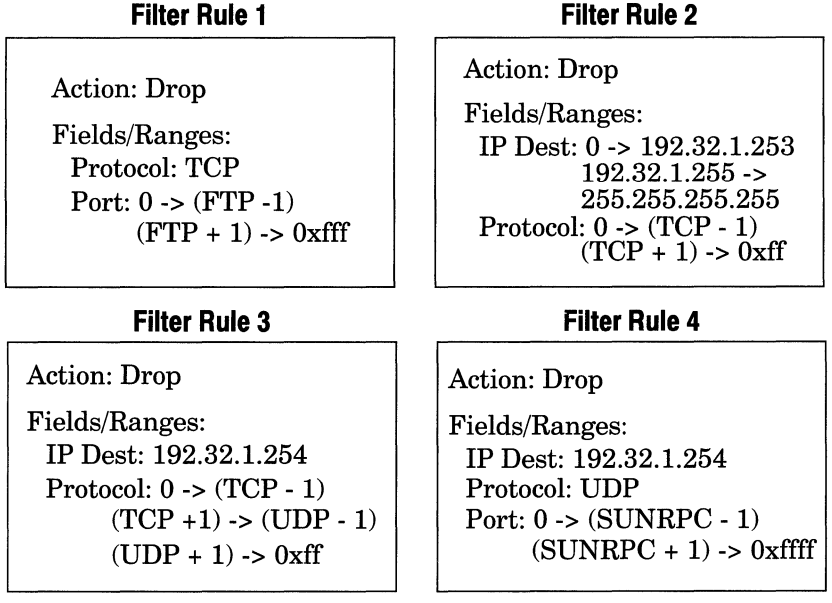


Figure C-4. Example of Version 7 Filters

Index

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

V

X

Z