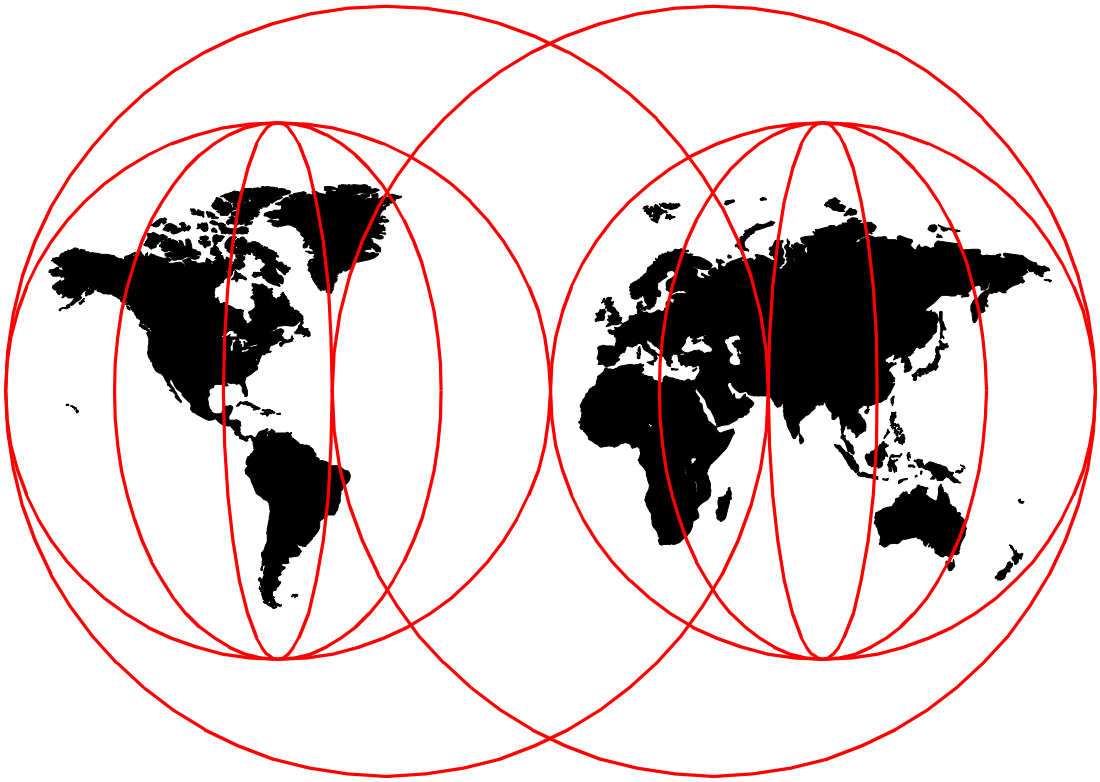
Layer 3 Switching Using MSS and MSS Release 2.2 Enhancements

Jonathan Follows, Jaap Burger, Jaime Garcia, Gary Norton



International Technical Support Organization

http://www.redbooks.ibm.com

SG24-5311-00

International Technical Support Organization

SG24-5311-00

Layer 3 Switching Using MSS and MSS Release 2.2 Enhancements

June 1999

Take Note!

Before using this information and the product it supports, be sure to read the general information in Appendix C, “Special Notices” on page 219.

First Edition (June 1999)

This edition applies to Version 2, Release 2 of Multiprotocol Switched Services Server code, which is a microcode feature for the 8210 Multiprotocol Switched Services Server and for the equivalent MSS Server Modules installed in the 8260 Multiprotocol Switching Hub and the 8265 ATM Switch. It also applies to Version 2, Release 2 of Multiprotocol Switched Services Client code which is a microcode feature for the Multiprotocol Switched Services Client adapter installed in the 8270 IBM LAN Switch and in similar devices.

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

Contents

Preface

This redbook describes the new functions of MSS Version 2 Release 2. With this release of code and the new ATM-attached Ethernet switches it is now possible to use the available ATM standards across IBM's network portfolio.

The first section of this book is a discussion of Layer 3 Switching: what it is and how it applies to ATM networks. The section ends with examples of practical implementations of Layer 3 Switching using IP, IPX, token-ring and Ethernet networks across an ATM backbone.

The second section of this book discusses the specific changes made in the latest release of MSS software and hardware, with examples where applicable.

The redbook will help you to implement Layer 3 Switching across ATM networks, especially with new hardware such as the 8371 Multilayer Ethernet Switch. It will also help you understand the concepts underlying Multiprotocol over ATM, why it is in many cases an improvement over Next Hop Resolution Protocol (NHRP) and is strategic in IBM's vision of networking using ATM.

The Team That Wrote This Redbook

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

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

Jaap Burger is an advisory IT Architect in IBM in the Netherlands. He has 13 years of experience in the field of wide area and campus networking. His areas of expertise include campus network design. He has written extensively on ATM and related standards and functions.

Jaime Garcia is a Product Engineer for ATM Networks. He holds a degree in Telecommunications Engineering and has four years of experience in the networking field. He has been working for Fringes Systemas Informaticos in Valencia, Spain, as part of the NHD Valencia PE team and providing field support to Europe, Middle East and Africa for IBM ATM networks. He is also involved in ATM-related courses taught at Fringes. His previous experience includes working in electronic systems design, also in Valencia, for two years.

Gary Norton is an advisory telecommunications specialist working for IBM Global Services in England. His areas of expertise include design and implementation of ATM campus networks, especially using MSS and token-ring.

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

Shawn Walsh, Erol Lengerli, Tate Renner, Gail Christensen
International Technical Support Organization, Raleigh Center

John Lloyd, Tonya Dorsett, Natarajan "Raj" Vaidhyanathan, Ed Rovner, Kevin Frick
IBM Networking Hardware Division

Mohammad Shabani
IBM NHD Education

James Forbes, David Winspear, Bernie Grunwald
IBM NHD Campus Technical Support

Gee Chia
IBM NHD

Comments Welcome

Your comments are important to us!

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

- Fax the evaluation form found in “ITSO Redbook Evaluation” on page 31 to the fax number shown on the form.
- Use the online evaluation form found at `http://www.redbooks.ibm.com/`.
- Send your comments in an Internet note to `redbook@us.ibm.com`.

Part 1. Layer 3 Switching

Chapter 1. Layer 3 Switching Standards

This chapter defines Layer 3 Switching and demonstrates its implementation in Asynchronous Transfer Mode (ATM) networks. This chapter is mostly a theoretical discussion of Layer 3 Switching rather than one discussing specific product implementations.

Some of this chapter gets reasonably technical in its discussion, especially in its discussion of the implementation of MultiProtocol Over ATM (MPOA). This may be interesting in itself but is also necessary to understand all the implementation options demonstrated in the second half of this book. However, if the reader chooses to skip the later sections in this chapter, it is hoped that the message "switching is good, and MPOA enables the implementation of switched networks across ATM" is understood.

As local area networks (LANs) grew, there was an ever growing need for more bandwidth which was partially satisfied by the implementation of LAN switches. The switches were used as replacements for shared hubs. Routers were used to transport data over relatively low-capacity wide area links by separating traffic and avoiding unwanted broadcast traffic. As the speed of both local area and wide area networks (WANs) has increased, routers have increasing difficulty in keeping up with the traffic volumes that they are handling. Layer 3 Switching is a solution that brings higher capacity to today's networks but still retains the vital routing *function* in the network. ATM opens up the possibility of constructing a network which deploys hardware *switching* of Layer 3 packets across both local area and wide area networks in order to bypass the performance constraints of router-based networks.

The number of protocols which are used in the local area network is also decreasing, with TCP/IP being the most important for the future. New generation high-speed networks will all have layer three switching capabilities for TCP/IP and other protocols such as IPX. In networks constructed around an ATM backbone, the entire ATM network will act as a single router image and will switch TCP/IP and other types of traffic directly between devices attached to the ATM backbone.

This chapter is in three sections. Section 1.1, "Evolution of Local Area Networking" on page 3 describes the techniques used for building networks from the first LANs up to ATM networks. Section 1.2, "MPOA: Merging Switching and Routing" on page 15 discusses the methods for building efficient switched networks. The chapter concludes with a specific discussion on how IBM implements standards and adds additional capabilities on top.

1.1 Evolution of Local Area Networking

This section is an overview of the changes in the way local area networks are implemented and why these changes were necessary.

1.1.1 Wires, Hubs and Switches

In the first instance, a local area network was simply a cable attaching all the users, as shown in Figure 1.

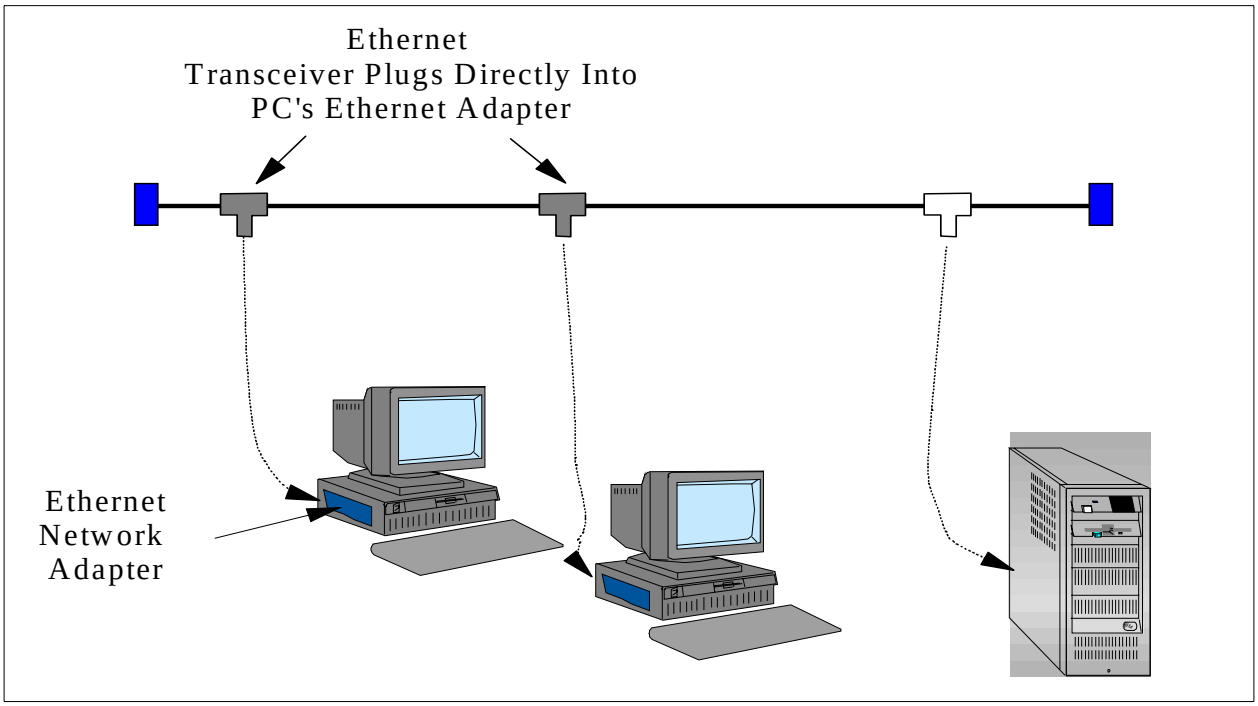


Figure 1. Early Ethernet Network

Networks like this had two major problems:

1. Each system needed its own tap into the cable. This was a process which had to be done at the place where the new workstation was situated.
2. If the cable broke, the entire network would be disrupted.

Hubs solved both of these problems; from every workplace a cable is installed from that workplace to a central point. Hubs still implement a single backbone connection which all users share, so the logical picture is still pretty much the same as for the cable of Figure 1, but hubs allow the addition and removal of users from the network without causing any disruption to the network.

Hubs have evolved from being relatively simple devices to today's hubs which are capable of monitoring the network, sending the information to a network management station, and even automatically circumventing problems which would otherwise disrupt the network.

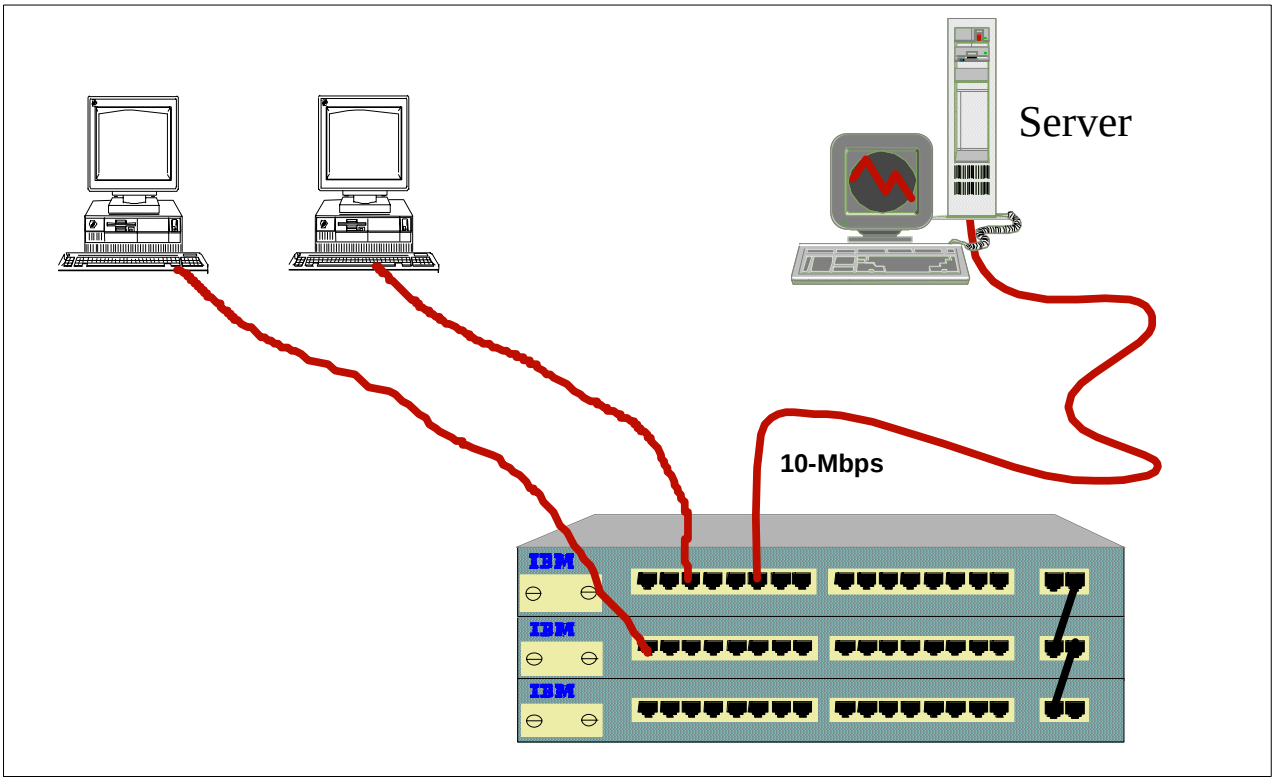


Figure 2. Wiring Concentration Using Hubs

Although shared hubs solved the major problems of manageability of the network, the shared environment could not always satisfy growing bandwidth requirements. Bridges could be used to split single LAN segments into several connected segments, but since bridges could only attach to two segments, a frame could traverse several segments before it reached its destination. A large, multi-bridge network rapidly became difficult to manage and, in any case, backbone LAN segments still had to cope with almost the entirety of the LAN traffic anyway.

Switches are essentially multi-port bridges in which frames are transported at media speeds between the switch ports. LAN switches allow multiple bridges to be replaced by a few switches, the advantage being that the switch has more ports allowing data frames to be switched directly from the originating segment to the destination segment. For devices which attach to a switch rather than to a hub, the key difference is that each device only receives frames which are destined to it, rather than having to receive all frames destined to all devices on the hub.

Although switches can be used simply to replace multiple bridges, the current comparative cost of LAN switches means that it is also feasible to connect many users directly to a switch port. Each user now gets dedicated bandwidth of 10 or 100 Mbps when Ethernet is used or 4/16 Mbps in the token-ring environment. Later implementations of LAN switches allow special *tricks* such as “full duplex” operation. Full Duplex operation is only possible between LAN switches and directly-attached devices with special LAN adapters; by using special code and breaking the normal rules it allows devices to send *and* receive at media speed over a single connection simultaneously, which is of particular benefit to heavy

network users such as servers. More recent switches support higher speeds such as 100 Mbps (Fast Ethernet) or even 1000 Mbps (Gigabit Ethernet). With end-to-end switching and using new technologies, like ATM or Gigabit Ethernet, in the backbone new ways of networking becomes possible.

The majority of users will be able to benefit from the higher speed of the network without having to change their network connectivity. The only change to this is that the cost of Fast Ethernet adapters means that in an Ethernet environment it is possible that increasing numbers of users have a dedicated bandwidth of 100 Mbps whilst 10 Mbps used to be normal some years ago.

1.1.2 Routers

The first personal computers were used as stand-alone systems; there was no need for them to be connected. Personal computers were also connected to central systems and ran software which emulated a host session running on a “dumb” terminal such as a 3270, but even then the other PC applications were still only for a single user with no need for LAN attachment.

Before long, though, this all changed because new applications were created which made use of a network, and the first shared local area networks were built. Over time the applications became more bandwidth-intensive and the number of attached users grew rapidly. The users on the network all had to share the available bandwidth, the bandwidth usage grew and before long the available bandwidth was not enough to satisfy all the requests. The result was that bridges were used to connect groups of users and each group had to share the bandwidth for that group. As we have seen, the technology has developed and now LAN switches are used to interconnect users and groups of users.

The next step in the local area network evolution was the connection of several locations, each having its own local area network. Even today, the cost of wide-area networking connections is significant, and it is not practical to provide wide-area connectivity with the same bandwidth as in local-area networks. Initial wide-area links between LANs were made over lines with a capacity of 64 kbps or lower, which compared to the capacity of the LANs themselves of up to 16 Mbps. Nowadays the same sort of difference still applies although the figures are different - there is still a significant difference between a T1/E1 (1.544 Mbps/2.048 Mbps) wide area link and a Fast Ethernet switched backbone running at 100 Mbps. This discrepancy means that only the traffic really destined to go across such a wide-area link should be allowed to make use of it.

In general when local area network users want to communicate they look for their communication partners by using a mechanism called *broadcast*, in which network users flood the network with searches for their partners. The first applications and protocols designed to be used in the local area networks relied on this broadcast mechanism. In a reasonably small network the number of broadcasts is not really a problem, but a network with many users and servers in a bridged configuration that relies on broadcasts for finding the session partners will create a large amount of broadcasts to the extent that the large numbers of broadcasts generated can consume a significant amount of the network’s capacity.

Bridges will generally transport the broadcasts between bridge ports since a standard bridge has no knowledge of the type of frames which are passing the

bridge. A router, on the other hand, has knowledge of the protocols it is to route, and if a protocol is not known to the router it will simply not transport that data. With the use of a router in a network the broadcasts cannot cross the router without the router having a look at them. The router will now take care of the address resolution. This will limit the impact of the broadcast to only the LAN segment where it is issued.

Routers, then, are used to subdivide a network into a number of broadcast domains and are essential for the transport of chatty LAN protocols across relatively low-capacity wide area links. Today the price of wide area networks has dropped and new techniques have made the usage of the available bandwidth more efficient, but in most networks the total price of WAN links (especially given that WAN links usually incur a repeating annual rental fee) will still be many times the cost of LAN hardware and the price per Megabit will remain many times the LAN price. The difference between LAN and WAN speeds also remains. Today we may use WAN connections with a capacity of the order of millions of bits per second, but LANs are now possible with capacities of thousands of millions of bits per second (Gigabit Ethernet, or ATM OC-48 at 2.488 Gbps) An efficient method of transporting data across the WAN remains very needed.

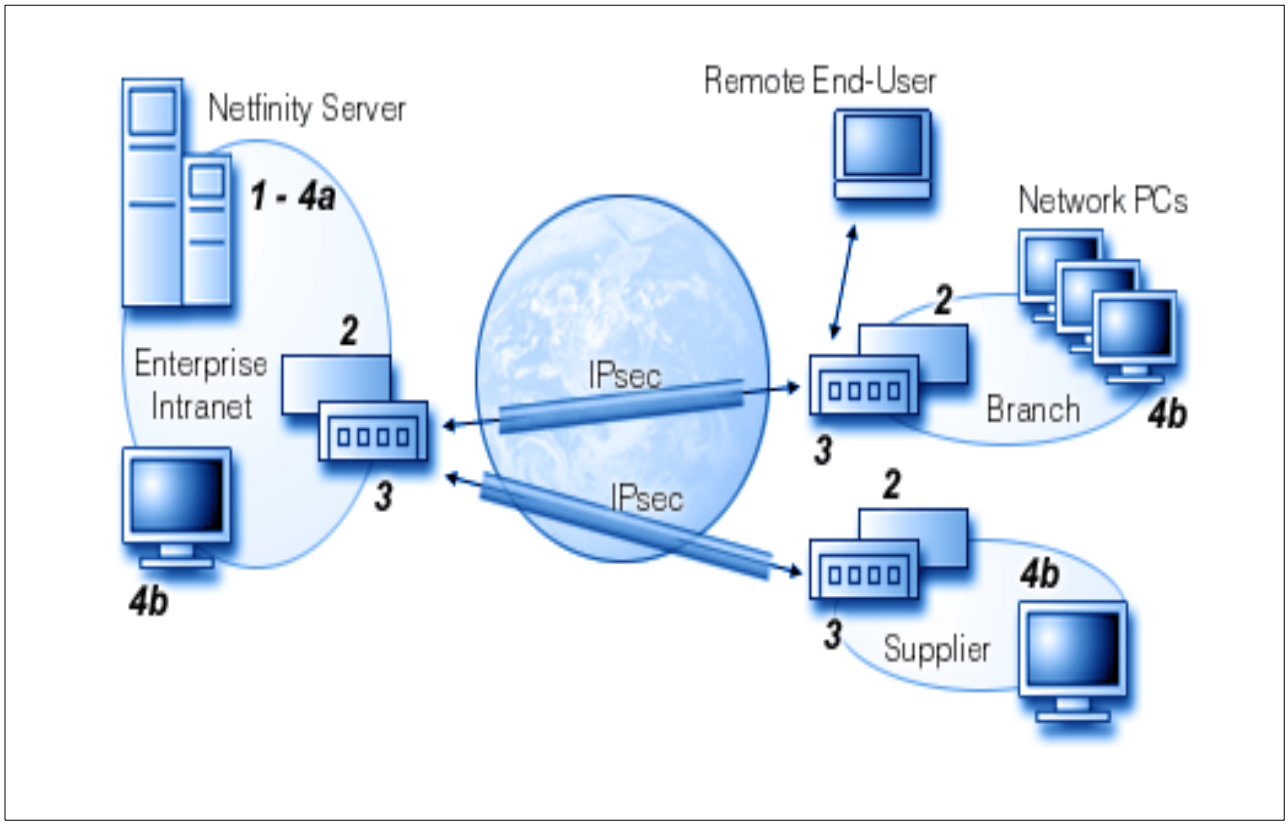


Figure 3. Routers in Wide-Area Networks Today

The whole evolution of the end user systems and servers as well as how people are now working together requires a network which has a high capacity and has the power to resolve addressing issues very fast. The nature of the application has also changed. Nowadays the network will not only be used for data applications, only requiring a certain amount of bandwidth, but also time-sensitive

applications like video and voice. The last two applications put very different demands on the network.

1.1.3 Moving to ATM

In 1.1.1, “Wires, Hubs and Switches” on page 4 and 1.1.2, “Routers” on page 6 the evolution of the networks is explained as well as the need for new technologies to cope with the ever growing bandwidth requirements of the users and the speeds the new technologies can provide.

One simplistic view of ATM is that it is a high-speed wide area networking technology. It is increasingly possible to buy high-speed public ATM wide area links, so that the bandwidth of the network in Figure 3 on page 7 can be increased by providing faster LAN links to the routers and using ATM to provide faster WAN links between the routers. This approach does have the advantage of simplicity.

The problem with this approach is that increasing bandwidth demands eventually make it impossible for routers to keep up with the speed of the network. The usage of routers in an ATM network makes the network expensive. Router interfaces and router engines that have to process ATM or Gigabit interfaces at wire speed are very costly, because each packet is examined for its destination address and forwarded. Further demands are made on router processing power when routers have to manage priority queues for traffic to be sent over congested links.

Figure 4 on page 9 shows the time a processor has per 53-byte ATM cell received on links of different speeds. On the left is shown the number of instructions a 10 MIPs processor would have per cell to make a decision. This figure clearly shows that wide area link speeds are increasing so fast that router processing power cannot keep up.

Layer 3 Switching is a solution to this problem. Layer 3 Switching refers to the ability of a network to *switch* layer 3 packets through hardware engines rather than *routing* the packets through software engines.

Going back to our simplistic network model in Figure 3 on page 7, Layer 3 Switching can be implemented in some or all of the routers themselves. However, the large router at the centre of the network is still expensive and a potential bottleneck: it may be possible to offload some of its processing power by implementing hardware Layer 3 Switching engines but this router still has to transport all the packets it receives on its wide area links onto its local area network connections, and vice versa.

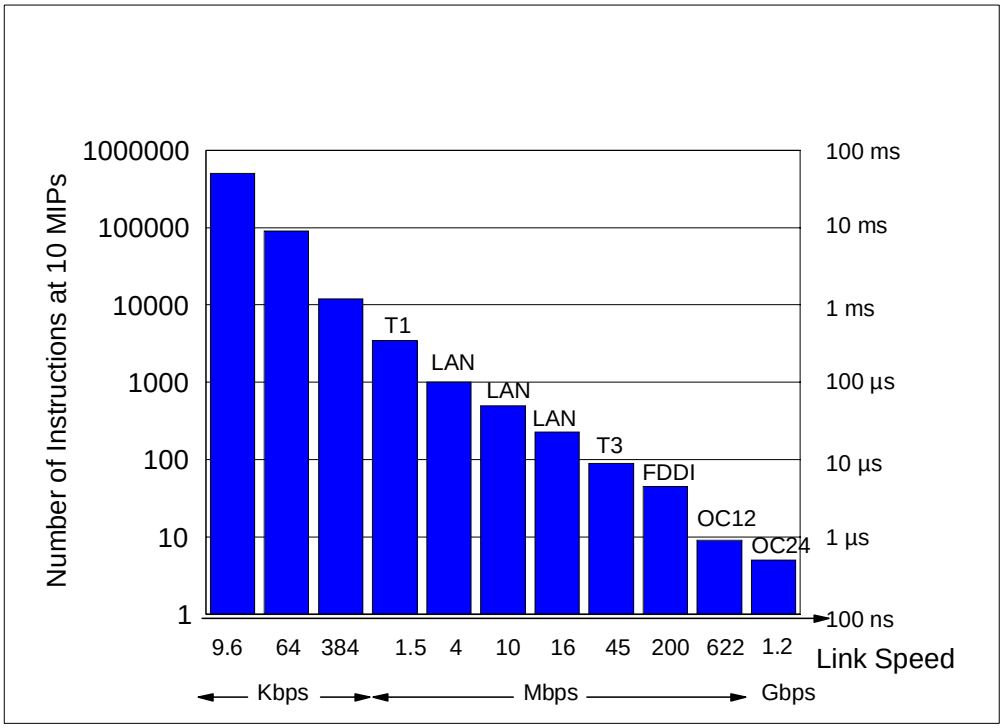


Figure 4. Line Speeds and Router Processor Capacity

ATM is a high-speed networking technology based on switching and can be implemented across local area *and* wide area networks. Standards such as Classical IP over ATM (CIP) or LAN Emulation (LANE) were developed to allow ATM to operate with existing LANs. But these standards also seem to bring with them the restrictions of existing LANs: Classical IP over ATM is only usable for systems using TCP/IP and the systems must all be directly connected to ATM; LAN Emulation directly emulates existing token-ring or Ethernet LANs. Communication with systems in different networks still requires routers in the ATM network.

A typical environment which uses ATM as a backbone technology has probably one or more of the following characteristics:

- A large network
- Users and applications with high bandwidth requirements
- Local and Wide area connections
- Requirements to merge multiple networks to a single network
- Voice and video requirements
- Most of the traffic has to go across the backbone

It is assumed that the reader is familiar with the concepts of both Classical IP over ATM and LAN Emulation. The following redbooks have been written to explain this in great detail:

- *MSS Release 2.1, Including the MSS Client and Domain Client*, SG24-5231
- *Understanding and Using MSS Release 1.1 and 2.0*, SG24-2115

- *Understanding and Using the IBM MSS Server*, SG24-4915

Because ATM is based on switching, it is now possible to design enhancements to CIP and LANE to use ATM switching throughout the network. The old saying “bridge where possible, route where necessary” can be updated to “switch where possible, route where necessary”, and ATM allows networks in which the vast majority of data traffic no longer needs to be processed by a router but can be switched in very fast ATM hardware switches.

One enhancement to the ATM network model, Next Hop Resolution Protocol (NHRP) is defined by the Internet Engineering Task Force (IETF) and is explained in 1.1.5, “NHRP” on page 11. The model for NHRP is *edge routing* as shown in Figure 7 on page 17. The major problem with NHRP remains the scalability of the router. All data must still go through a router. Figure 4 on page 9 shows that eventually a router does not have the processing power to route the data at wire speed.

So a new model had to be invented. The ATM forum has taken the good parts of several standards and based on this it has created the Multiprotocol over ATM (MPOA) standard. MPOA is based on LAN Emulation Version 2 and Next Hop Resolution Protocol. In the MPOA standard NHRP is modified in such a way that it is usable for a LAN Emulation environment. MPOA is explained in 1.2, “MPOA: Merging Switching and Routing” on page 15.

1.1.4 Layer 3 Switching Defined

Layer 3 Switching is the action of a network device in switching data packets in hardware based on their Layer 3 protocol information, for example the destination IP address. Without this function these data packets would have to pass through the routing engine of a router.

A **Layer 3 Switch** is a network device which is a Layer 3 device (in other words, a router) but which has the capability of performing Layer 3 Switching between its network interfaces.

These definitions lead to the following observations:

1. The 8371 Multilayer Ethernet Switch is capable of Layer 3 Switching because it can identify flows at the Layer 3 IP layer and switch IP packets between its ports directly. It is not a Layer 3 Switch because it does not currently act as a router, and therefore an external router is still required in order to initiate data transfer between devices on different IP networks and to participate in all the normal “router” protocols such as RIP and OSPF.
2. An entire ATM network can be considered as a Layer 3 Switch if the central routing engines implement the MPOA Server function and the edge devices implement the MPOA Client function, because in such a network Layer 3 packets (currently IP and IPX) will be switched across the network in hardware (between the “interfaces” which are now the *ingress* and *egress* MPOA clients) and will not pass through any software routing engines.

The following network diagram shows a network based on an ATM backbone in which the only routers are in the core of the ATM network, yet almost all[1] the

[1] The reason for the use of the world “almost” is twofold: MPOA doesn’t establish shortcuts until a *flow* of traffic has been identified but also because devices in the network which are not running as MPOA Clients, such as the PCs numbered “1” and “6”, are not themselves capable of setting up direct shortcuts across the ATM network. Even for these devices, though, MPOA is used by their nearest router to cut out all but one actual routing hop.

traffic which flows between devices in different IP networks can be switched over ATM hardware without having to pass through either of the routers.

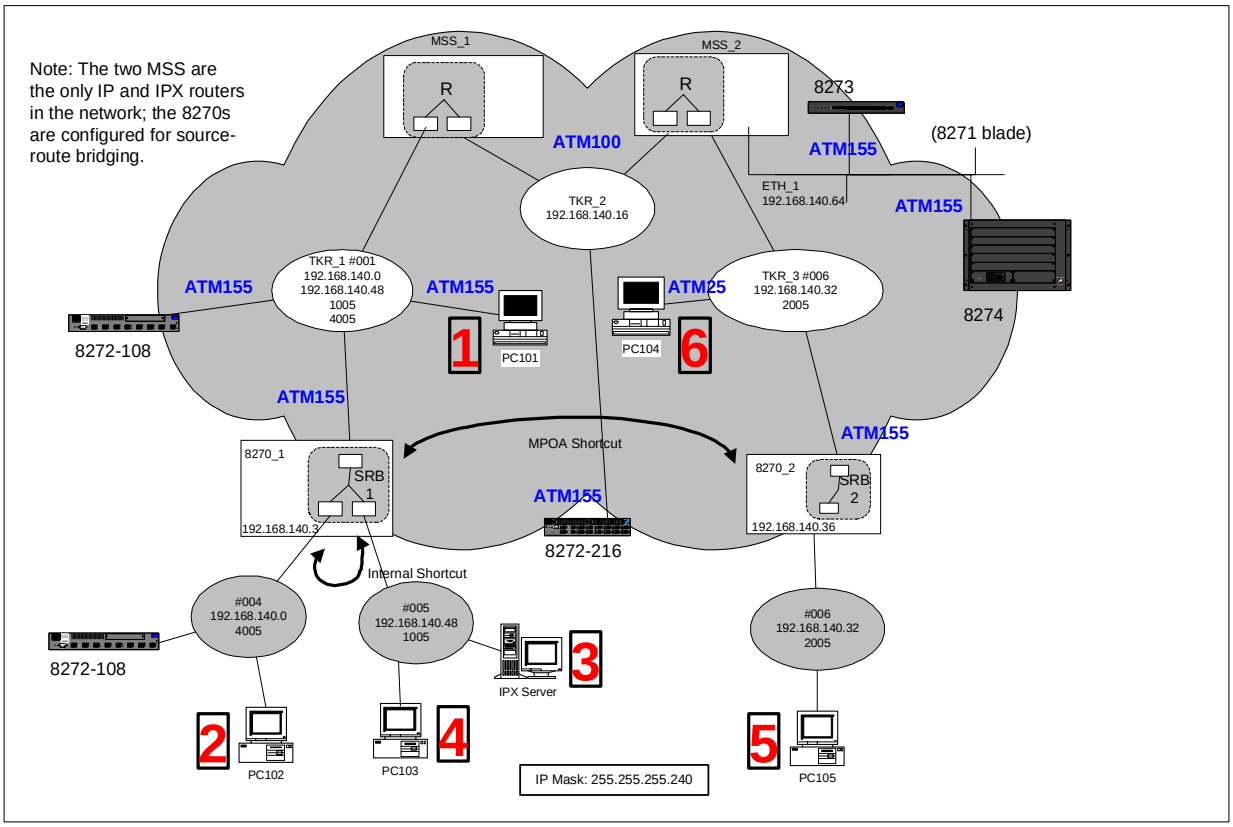


Figure 5. Layer 3 Switching Implemented in an ATM Network

1.1.5 NHRP

The Next Hop Resolution Protocol (NHRP) provides a mechanism for ATM-attached edge IP routers to set up direct ATM shortcuts to other ATM-attached routers. NHRP allows routers to take advantage of the fact that the underlying ATM infrastructure allows direct connections to be set up between them, even when these routers are logically in different IP networks. This direct connection is then used for actual data transmission, bypassing one or more intermediate router hops, reducing the load on the network and increasing throughput through the network. As defined in RFC 2332, the ATM Forum Standard for NHRP is only applicable to Classical IP networks; IBM additionally implements a super-set of RFC 2332 which allows shortcuts to be established to and between LANE networks.

ATM and frame relay networks are examples of networks which are known as Non-Broadcast Multi-Access (NBMA) networks in which it is not possible to find communication partners by using a broadcast mechanism, yet multiple devices connect to a single network. The Non-Broadcast Multi-Access (NBMA) network communication partners must know the address of the other side of the network before any communication will take place. In an IP network, some mechanism must be in place to resolve the ARP (Address Resolution Protocol) requests which will flow across the network. This is the purpose of the ARP server function; Classical IP cannot be extended between subnets so that in a

configuration with more subnets routers are used for communication between the subnets.

The approach used by the NHRP protocol is one of having a client/server setup in the NBMA network. The NHRP Servers (NHS) are implemented on routers in the core of the network; NHRP Clients are ATM-attached edge routers. The servers maintain NHRP Client (NHC) information elements (CIE) for all clients which they are observing. The protocol defines mechanisms for the server to collect information directly from the clients and to exchange information among themselves about the total network topology. A client in need of information sends a request to the NHRP server and gets a response containing the CIE.

The NBMA network can be divided into independent logical IP subnets (LIS) connected by routers. All the hosts in a single LIS have the following properties:

- All nodes have the same IP network number and IP netmask
- All nodes of a LIS belong to the same NBMA network
- All nodes outside the LIS are accessed through a router
- All communication between nodes in the same LIS goes directly between them on an ATM connection (PVC or SVC)

This means that, without NHRP, communication between LISs must be through routers. These routers must be connected to two or more LISs for the forwarding between these LISs. Of course this creates overhead on the router and a higher latency through the network.

NHRP is defined in RFC 2332 as an extension to the existing Classical IP specification RFC 1577. This means that RFC 2332 is designed only for a Classical IP environment, but the specification also allows for vendor-specific extensions to make NHRP suitable for LANE environments as well. IBM has a NHRP implementation which allows NHRP to be used with LAN Emulation. This makes it possible to create shortcuts between:

- Networks that consist of LIS only
- Networks that consist of ELANs only
- Networks that consist of both LISs and ELANs

In Figure 6 on page 13 an overview of the main components NHRP is shown along with some of the control and data traffic flows across such a network.

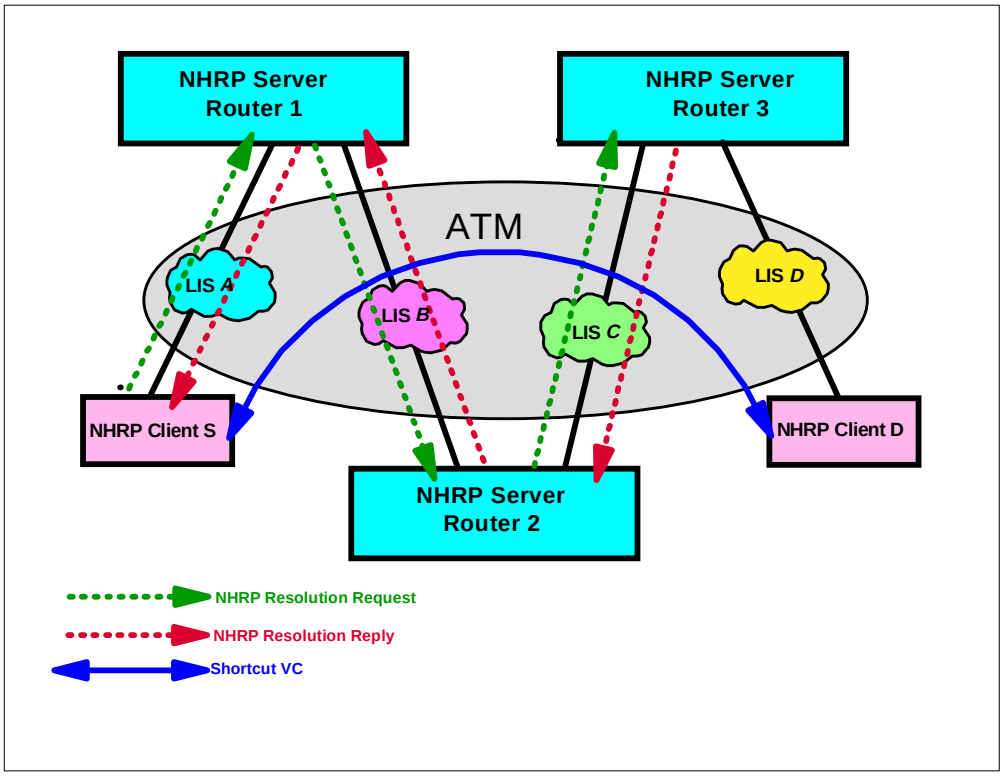


Figure 6. The Main Components of NHRP and the Logical Flow

NHRP has two major components. The NHRP Server (NHS) performs the Next Hop Resolution Protocol Service in the network. The NHS is always associated with a router. Each LIS needs an NHS for address resolution. All the NHSs in an NBMA network act together to resolve the path across the network.

The NHRP Client is a station which has no NHS functions at all. It is either an ATM-attached host or an ATM-attached router, and will typically be the latter. It has to register at its NHS to allow the server to be able to participate in the process of setting up shortcuts: a resolution request never goes all the way to the destination NHC but is resolved in the serving NHS. Each NHC keeps a cache of all the resolved addresses returned to it; in the standard this can be done either automatically or manually but IBM's implementation does not allow for any manual configuration.

The result of the NHRP resolution process is that an NHRP Client has established a direct switched path across the ATM network to another NHRP Client. Data traffic between these devices will now flow using a switched, hardware path, and will not pass through the intermediate routers in the ATM network.

The following NHRP protocol packets have been defined in RFC 2332:

- The NHRP Resolution Request is sent by the NHC to its serving NHS when the NHC wants to discover the ATM address of a destination NHC. When the request arrives at the NHS which serves the destination (and therefore knows its ATM address) it is able to generates a positive NHRP Resolution Reply.

The NHRP Resolution Reply packet is then sent back by each NHS in turn until it reaches the requesting NHC.

- The NHRP Registration Request is used by a station to register at the NHS.
- The NHRP Purge Request is sent by the NHS to stations to delete previously cached data. This is primarily done if that data is no longer valid.
- The last NHRP Protocol Packet is the NHRP Error Indication. This packet is sent to the originator of the initial packet to tell the sender about a possible error in its packet.

In Figure 6 on page 13 NHRP Client S wants to communicate with NHRP Client D. If the ATM address of NHRP Client D is already known to NHRP Client S it will use that information to set up a direct connection. If the ATM address of D is not known, NHRP Client S will construct a NHRP Resolution Request containing information about D's IP address and both the IP address and the ATM address of NHRP Client S. S may indicate that it only wants an answer from the NHRP Server which actually serves D**,** or that intermediate NHRP Servers which have knowledge of D may also answer. The difference is that a response from the NHRP Server which actually serves the destination NHRP Client is called *authoritative*, whereas other replies are called *non-authoritative*. In our example, the request is processed by the next NHRP Server in the path, Router 2, and since the ATM address of the destination NHRP Client is not known to this server it will be forwarded to the next known server. The NHRP Server on Router 3 serves NHRP Client D, so when the request reaches this server it can return the NHRP Resolution Reply with the required data. The resolved resolution request may be cached at the intermediate NHRP Servers for future use in responding to non-authoritative requests. S can now establish the shortcut virtual channel connection (VCC) to D.

IBM has made several extensions to NHRP to enhance its usability. The formal standard only allows shortcuts to be set up between NHRP Clients, so if the source is a client and the destination is not, the standard does not allow the creation of any shortcuts. IBM's NHRP Server implementation in the MSS can send an NHRP Resolution Reply which contains the ATM address of a destination which is not an NHRP Client.

The standard specifies RFC 1483 encapsulation over ATM both for NHRP control flows as well as on the actual shortcut VCC. According to the standard, this means that no shortcut VCC can be established between stations that are attached to ATM using LAN Emulation; the MSS implementation of NHRP supports LANE encapsulation and LANE shortcut VCCs by extending the NHRP standard.

These vendor private extensions are allowed for in the standard; how the extensions are implemented is up to the vendor.

In addition, because IBM's implementation of NHRP allows a shortcut to be set up to a device that is not an NHRP client, if a network is partitioned into an area of NHRP Servers and non-NHRP Servers, the last NHRP Server implementing the IBM extensions will provide the ATM address of the next hop router. This allows the establishment of a shortcut from the origin to the non-NHRP router.

In the previous paragraph the IBM extensions to NHRP were mentioned; as we will also see with MPOA, IBM conforms to the published standards but also provides “extensions” to enhance the usability of NHRP enormously.

In the IBM LAN Emulation environment, MSS Servers use the IBM extensions to provide NBMA information (ATM address information and MAC address information) for devices on their ELANs. As part of these extensions, a LAN Emulation Shortcut Interface (LSI) is automatically created for each physical ATM interface when NHRP is enabled. The LSI is different from a traditional LAN Emulation Client (LEC) interface because it provides more than one LAN encapsulation type concurrently. So on a single LSI there may be a VCC using token-ring encapsulation whilst another VCC is using IEEE 802.3. The LSI can also connect to more than one Emulated LAN. LSI connections are always unidirectional and can only transmit data, but never receive.

1.2 MPOA: Merging Switching and Routing

Multiprotocol over ATM (MPOA) provides a mechanism for ATM-attached devices to set up ATM shortcuts to other ATM-attached devices. MPOA Clients, the name for these ATM-attached devices, differ from NHRP Clients because they do not implement Layer 3 routing functions[2] such as IP routing, but only need to be able to identify Layer 3 addresses (such as the destination IP address for IP packets) and relate these to the ATM address of the destination ATM address (such as the address of a Layer 2 switch). MPOA allows actual data transfer across the ATM network to bypass the routing engines entirely.

IBM’s current MPOA implementation allows these shortcuts to be set up for both IP and IPX traffic flows. MPOA is typically implemented in LAN switches with ATM uplinks but the standard allows other devices such as directly-attached ATM hosts to implement MPOA; although few implementations exist today there is nothing to stop hosts such as IBM S/390 mainframes from implementing MPOA in the future. Just as for NHRP, IBM complies fully with ratified standards and implement extensions which increase the capabilities of IBM’s MPOA Clients in a network, such as the ability for an MPOA Client such as the 8371 Multilayer Ethernet Switch to set up a shortcut to a standard LANE client which has no knowledge or understanding of MPOA (which might be very useful if, for example, we are talking about a large S/390 server configured as a LANE client).

Table 1. Some Differences between NHRP and MPOA

	NHRP	**MPOA**
Client Host	IP Host in CIP network or in LANE network (IBM extensions)	MultiProtocol Host in LANE network
Client Edge Device	IP Router with ATM uplink	Switch with ATM uplink

[2] The only reason for configuring an IP address in an MPOA Client Edge Device will be for network management. The IP addresses in the diagrams which follow have been configured solely for that purpose.

	NHRP	MPOA
Server	IP Router with ATM uplink	MultiProtocol (IP, IPX) router with ATM uplink
View from LAN device	Edge device is IP router	Edge device is bridge (for example, SRB for token-ring)
Standards	CIP	LANE
Model	Edge router	Edge switch, virtual router
LAN media	Routing, so not an issue	Shortcuts between token-ring and Ethernet if desired
IBM Extensions	LANE, shortcuts to non-NHRP devices	Shortcuts to non-MPOA devices, hardware path in egress MPOA Client

In a joint effort the IETF and the ATM Forum have created a standard which fulfils the requirements for a high capacity network. The Multiprotocol over ATM standard defines a network that is both capable of handling high speed connections and also one which makes full use of a switched environment. The standard handles multiple network layer protocols and mixed physical infrastructure. Packets are switched without a router in the data path.

The standard is also described as the *virtual router model*. In comparison with a router, the MPOA Server can be compared with the processor of a router, the ATM switching fabric is the backplane of a router and finally the entry points into the ATM network, or MPOA Clients, provide similar functions to a router's interface card. Figure 7 on page 17 gives an overview of the difference between the two models. The Edge Router model does not really need ATM as a backbone infrastructure. Every type of connection may be possible, but ATM gives distinct advantages compared to leased lines. In an ATM backbone we can define several virtual connections while in a leased line configuration the connection will only be to one other router. For several connections we need more adapters and lines to make these connections.

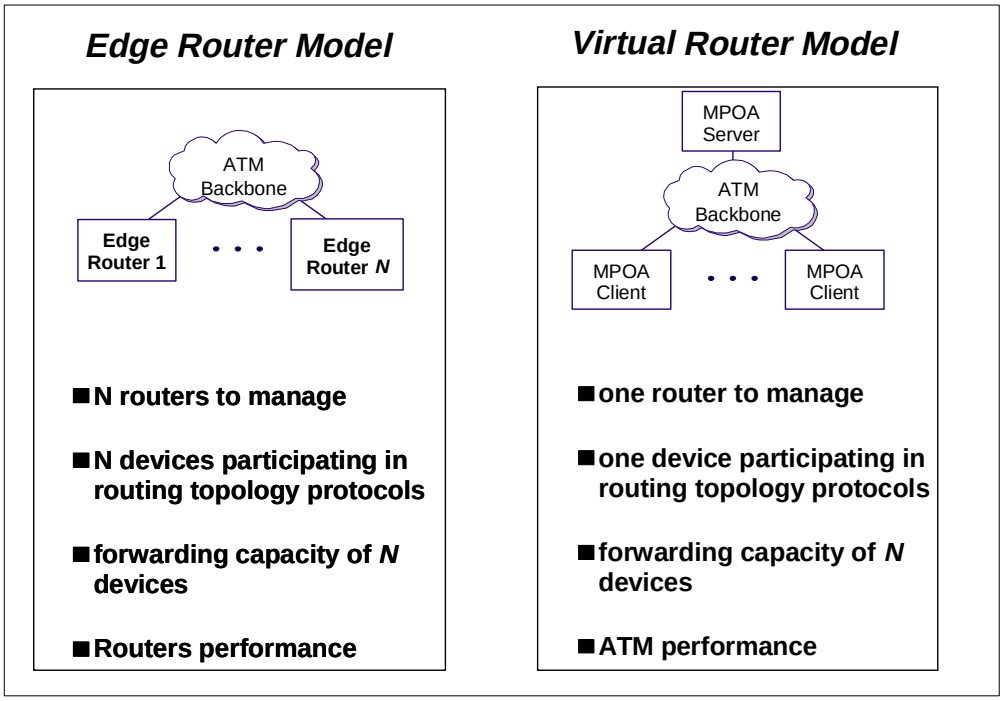


Figure 7. The Edge Router and Virtual Router Models

Initially with MPOA each session is established like all other sessions across an ATM network. The data path is made across the MPOA servers to the communication partner in the network and, just like a normal routed network, if multiple routers exist in the data path, the data connection will pass through each one of them. Only if the sender exceeds a (configurable) threshold of transmitted data rate is the MPOA Client triggered to create a shortcut directly to another MPOA Client in an attempt to bypass some or all of the routing hops. Now a direct path is created between the two clients and data can be switched directly between these clients. This creates a very fast data path. The switching between the two clients will be done in hardware and if the switches have enough capacity they will switch at wire speed.

Figure 8 on page 18 is a conceptual example of an MPOA-enabled network. In 1.2.2, “The MPOA Standard” on page 20, MPOA is explained in great detail. Figure 8 shows the two basic flows. The default path is comparable with the data flow in a standard LAN Emulation or Classical IP (CIP) network, passing through intermediate routers. The MPOA servers in the picture are in ATM-attached routers, but because of ATM there is the ability to set up a single switched hardware ATM connection directly between the ingress and egress MPOA clients. In an MPOA-enabled network the default path will be used for data flows with a low bandwidth utilisation. A soon as the number of frames is high enough that a certain utilisation threshold is met, the shortcut will be created and used for the direct data connection between the MPOA clients.

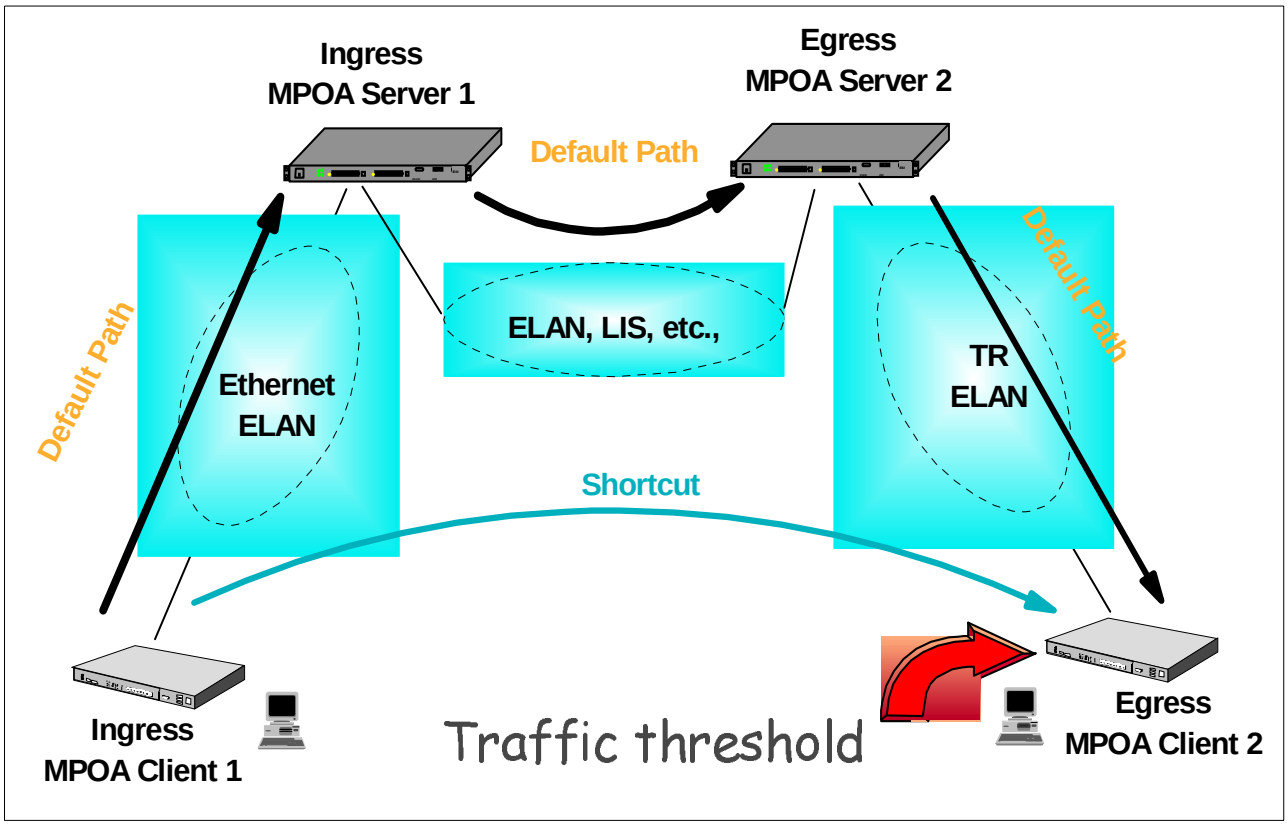


Figure 8. Data Flow in an MPOA-Enabled Network

Figure 8 demonstrates terms we will mention frequently in this book:

- The ingress MPOA Client is the ATM edge device which monitors Layer 3 traffic and detects *flows*. When it detects that traffic to a specific L3 destination has exceeded a pre-configured rate it attempts to set up a shortcut across the ATM network to bypass one or more routers in the network.
- The egress MPOA Client is the ATM edge device to which shortcuts are established. According to the standard, it has to substitute information into the data link (DLL) header of packets received over these shortcuts in order to transmit them over a LAN interface so that the packets appear to have followed the “default path” shown in Figure 8.
- The ingress MPOA Server receives requests from the ingress MPOA Client to initiate a MPOA shortcut.
- The egress MPOA Server provides instructions to the egress MPOA Client to allow the latter to receive packets over a MPOA shortcut.

In general both MPOA and NHRP offer wire speed routing functions. MPOA has some advantages if more than just IP and Ethernet are used, since MPOA can route more than just IP and also switch between Ethernet and token-ring. The backbone of the ATM solution is more robust. Using PNNI between the ATM switches and having more than one link between switches will provide a higher degree of redundancy than a layer 3 switching backbone. Rerouting of data is possible without the user even noticing it or without Ethernet even noticing it.

1.2.1 Migrating to ATM and MPOA Networks: The Virtual Router Model

In 1.1.2, “Routers” on page 6 the demands for a high capacity network were explained as well as some of the approaches being used to satisfy the bandwidth demands. One of the possibilities is the deployment of ATM in such a network.

Initially ATM was used for small environments and where each user had a need for a lot of bandwidth. The first ATM implementation was *Classical IP over ATM* (CIP). The IETF standard, RFC 1577, describes a method for attaching systems directly to an ATM switch. It provides a way for IP devices with ATM interfaces to communicate with each other across the ATM switched infrastructure. All the systems are part of the same subnet and for communication outside this subnet a router is needed. The benefit of using Classical IP over ATM was that it gave the users access to high-bandwidth connections between them. Other than bandwidth, no other advantages specific to ATM are used in this standard. The standard[3] says:

“This memo defines an initial application of Classical IP and ARP in an Asynchronous Transfer Mode (ATM) network environment configured as a Logical IP Subnetwork (LIS) as described in Section 3. This memo does not preclude the subsequent development of ATM technology into areas other than a LIS; specifically, as single ATM networks grow to replace many Ethernet local LAN segments and as these networks become globally connected, the application of IP and ARP will be treated differently. This memo considers only the application of ATM as a direct replacement for the “wires” and local LAN segments connecting IP end-stations (“members”) and routers operating in the “classical” LAN-based paradigm. Issues raised by MAC level bridging and LAN emulation are beyond the scope of this paper.”

The ATM Forum created a standard for a more general use. *LAN Emulation* allows emulated LANs to be created across ATM networks to which existing users can be connected. These LANs can emulate either token-ring or Ethernet. Users with very high bandwidth demands can connect directly to the ATM network and use their emulated LAN connections as high-speed versions of existing LANs; many users will not connect directly to the ATM network but instead connect via a *proxy* device such as a LAN switch with an ATM uplink. For the applications nothing really had to change and the available bandwidth was increased. LAN Emulation is now a fully mature standard. Now it is possible to have a network that is switched, ATM , and users can connect to it, LAN Emulation.

The problem with LAN Emulation on its own is primarily the lack of scalability of such a network. It is a perfect solution for a network in a single building, but it is not easily usable in a large campus environment. Just as in a large LAN network, the respective Emulated LANs will probably be connected with ATM-attached routers, and provided that the traffic demands on the routers in a data-only network are not too great, this can be a viable solution. But for networks which carry more than data or carry increasing volumes of time-sensitive critical data traffic no guarantees can be given for issues like latency, and using traditional routers can have a serious performance impact. All data will go through the router and the capacity of the router for adequate operation must be adequate and in proportion to the sum of the line capacity to which it is connected. For example, this means that a router must be capable of routing between two OC-3 (155 Mbps) interfaces. OC-3 is now a much used speed, but higher speeds are already

[3] RFC 1577 *Classical IP and ARP over ATM*, superseded by RFC 2225 with the same title and similar introduction.

in use like OC-12 (622 Mbps) or OC-48 (2.4 Gbps). Traditional routers will find it increasingly difficult to keep up with these speeds.

1.2.2 The MPOA Standard

Using the MPOA Virtual Router Model has two major benefits:

- Scalability
  - Route calculation capacity can be increased by adding more MPOA Servers (MPSs)
  - Forwarding capacity can be increased by adding more MPOA Clients (MPCs)
  - Backplane capacity can be increased by adding more ATM network capacity
- Manageability
  - Single virtual router image
  - Auto-configuration
  - Dynamic device discovery protocols
  - Reduced complexity of edge devices by eliminating the need to preform route calculation in them

The Virtual Router model provides a cost-effective foundation that can scale to meet the networking needs of the future. It provides scalability and manageability benefits and allows performance to be increased in an incremental vendor-independent manner. For example, forwarding capability can be increased simply by adding more MPCs. Similarly, backplane throughput can be increased with the addition of ATM network capacity. It is easier to manage the single virtual router image provided with MPOA than to deal with the complexities of administering multiple distributed edge routers participating in routing topology protocols. Additionally, MPOA includes auto-configuration and device discovery protocols that minimise device-specific configuration.

In this section we explain the function of the MPOA Virtual Router model.

The initial deployment of ATM was in primarily in networks which only covered a single building. The connection between buildings was generally made by routers across the wide area network.

For this type of network the LANE implementation is sufficient if the number of ATM-attached stations is not too big and ideally we have only one type of LAN, either token-ring or Ethernet. Communication between different ELANs is by using routers.

Next Hop Resolution Protocol, as described in 1.1.5, “NHRP” on page 11, was the first attempt to extend the usability of ATM with the addition of IP address resolution which goes beyond the logical IP subnet. The standard only allows you to use NHRP in the Classical IP environment, but IBM extensions made NHRP also usable for LANE and for connecting non-NHRP stations.

NHRP is used only in the IP network environment and requires that shortcuts be set up by ATM-attached IP routers. In fact the standard mandates that NHRP must be used between IP routers also. MPOA allows shortcuts to be set up between any ATM-attached device.

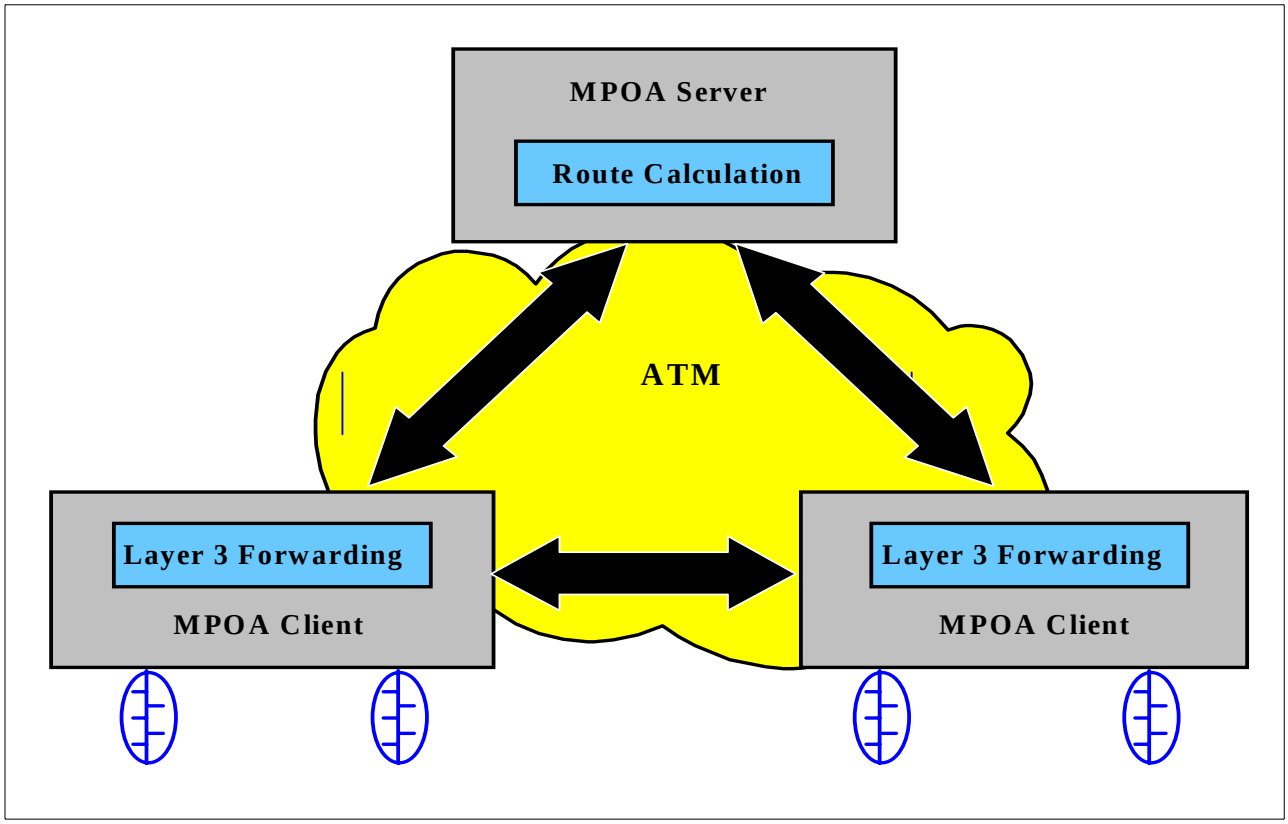


Figure 9. Multiprotocol over ATM (MPOA) Basic Functions

MPOA uses three techniques to do its work. These are:

1. ATM Forum's LAN Emulation (LANE),
2. IETF's Next Hop Resolution Protocol (NHRP)
3. The concept of the Virtual Router

LANE supports native LAN environments over ATM in a transparent manner, while NHRP provides a mechanism to establish a shortcut over the ATM backbone based on network layer addressing. NHRP is only used in the ATM backbone as a mechanism for deriving shortcuts and is not implemented by the MPOA Client itself. Virtual Routers provide the ability to separate functions among various elements of the network, which reduces cost and improves efficiency.

MPOA overcomes some of the performance and scalability limitations of LANE-based networks. LANE Version 2 is an integral component of MPOA. LANE is used for intra-subnet communications, while the MPOA virtual router provides communications between subnets.

The Virtual Router is a set of devices operating over a network which together provide the functions of multiprotocol routed networks. In the case of MPOA the edge devices are analogous to router interface cards; the ATM switching can be compared to the backplane of a traditional router, and the MPOA Server can be seen as the control processor. The MPOA framework defines the protocols

between the MPOA Server and the edge devices that enable the virtual router model.

The MPOA model distributes routing among edge devices and ATM-attached hosts with MPOA Clients, which forward packets, and MPOA Servers, which supply routing information. MPCs examine both the destination MAC address[4] and the destination layer 3 address of packets received on legacy LAN segments in order to make the correct forwarding decision. If the packet is to be routed, it will contain the destination MAC address of the MPOA Router interface. If so, the MPC will look at the destination network layer address of the packet, and resolve this to the correct ATM address based on the information received from the MPOA Server or from information in its cache. The MPC will, if necessary, then establish a direct virtual channel connection (VCC) to the appropriate destination. The destination ATM address can be the address of the destination host, if ATM attached, or the address of the device through which the packets should be forwarded, such as an ATM-attached LAN switch.

If the MPOA Client does not have any information for a destination ATM address for a particular packet, it forwards it on the “default path” as shown in Figure 8 on page 18 although it may also decide to initiate the attempt to set up a direct ATM connection for future packets it receives for the same destination.

If the packet is destined to a host in the same subnet, it can be bridged and the MPC will use standard LANE to resolve the ATM address and establish a virtual channel connection directly to the destination.

If the local MPOA Server does not know the appropriate ATM address when requested for information for a particular destination Layer 3 address, it can propagate the query to the other MPOA Servers or routers using NHRP requests. MPOA works at network Layer 3 to recognize the beginning of a data transfer and respond with a network route destination address. The shortcut VCC is then used to forward traffic using standard Layer 2 switching. With both Layer 3 and Layer 2 capabilities, the MPOA model encompasses routing and switching:

1. Being able to route and switch network layer traffic and
2. Being able to bridge non-routable traffic

The network layer mapping enables QoS properties of ATM to be used by network applications. For example, the IETF’s RSVP protocol operates at the network layer, and provides mechanisms for applications to reserve a particular quality of service. The MPOA framework allows the Layer 3 reservations to be mapped onto the underlying ATM fabric.

The fundamental concept behind the use of MPOA to support multiprotocol LAN-LAN traffic is based on the fact that, in most cases, data transfer usually occurs in a relatively steady flow. That is, a file or message being sent usually consists of multiple frames. For example, a 45KB file, using a typical Ethernet frame size of 1500 octets would require about 30 frames. Since all 30 frames would travel to the same destination, it is possible to identify the destination and establish an SVC based on the information contained in the first frame. Then all 30 frames could be broken into approximately 900 ATM cells and transmitted over the virtual channel established by the SVC. This could be considered a shortcut in that the entire flow of data follows a pre-established path, avoiding the default

[4] In fact, since the MPOA client is connected to the bridge function of a LAN switch, it could be argued that the MPOA Client already “knows” the source and destination Layer 2 MAC information before it starts its work.

path followed by routed traffic, and greatly improving performance. In the case of steady stream transmissions such as video, this is highly efficient and superior to simple router to router operation.

To set up a direct shortcut connection, MPCs obtain the ATM address of the exit point to which the destination host is connected. The destination host is a host with a network layer address that is either connected to a legacy LAN or is ATM attached. If it is a host connected to a LAN such as Ethernet or token-ring, the MPS returns the ATM address of the edge device that connects to the host on the LAN. If the host is ATM attached, the MPS returns the ATM address of the host that corresponds to its network layer address.

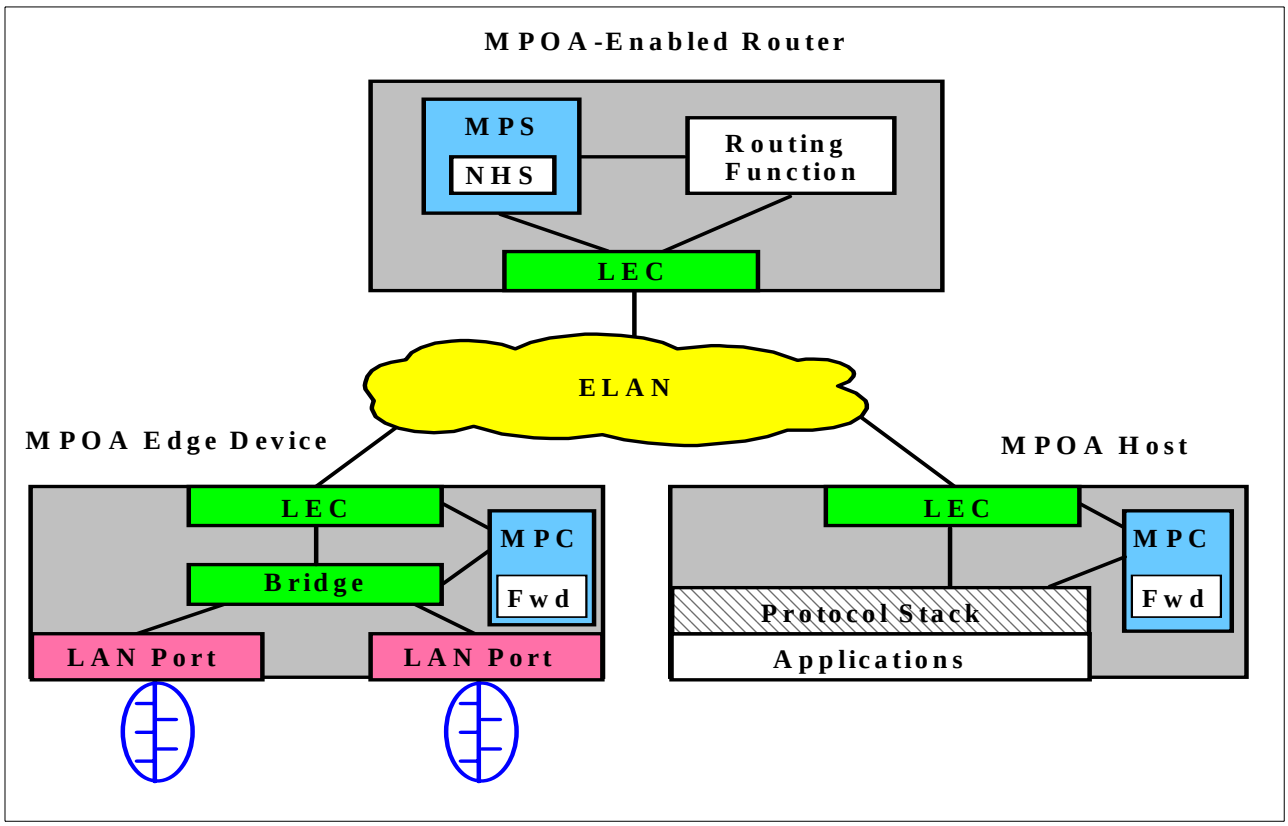


Figure 10. Multiprotocol over ATM (MPOA) Components

The events that allow a packet to be sent across an MPOA network using the shortcut capabilities of the MPOA system are:

- The packet enters the MPOA system at the ingress MPOA Client (MPC). By default, the packet is bridged using LANE to the default router. From there it is forwarded via the router in the MPOA Server to the destination edge device or another router. However, if this packet is part of a flow for which a short-cut has been established, the ingress MPC strips off the Layer 2 encapsulation from the packet and sends it via the shortcut
- If no data flow is detected, each packet being sent to an MPOA Server is tallied by its Layer 3 destination address as it is being forwarded by LANE. When the threshold is exceeded (“N” packets to a specific Layer 3 address within “X” time), the MPOA Client sends an MPOA resolution request to the

MPOA Server to obtain the ATM address to be used for establishing a shortcut to the Egress (exit) MPC.

- On arriving via a shortcut at the Egress MPC, the packet is examined and either a matching Egress Cache Entry is found or the packet is dropped. If a match is found, the packet is re-encapsulated using the cache information and it is forwarded via a LAN interface to its destination.
- The shortcut is an ATM SVC established for the specific data flow

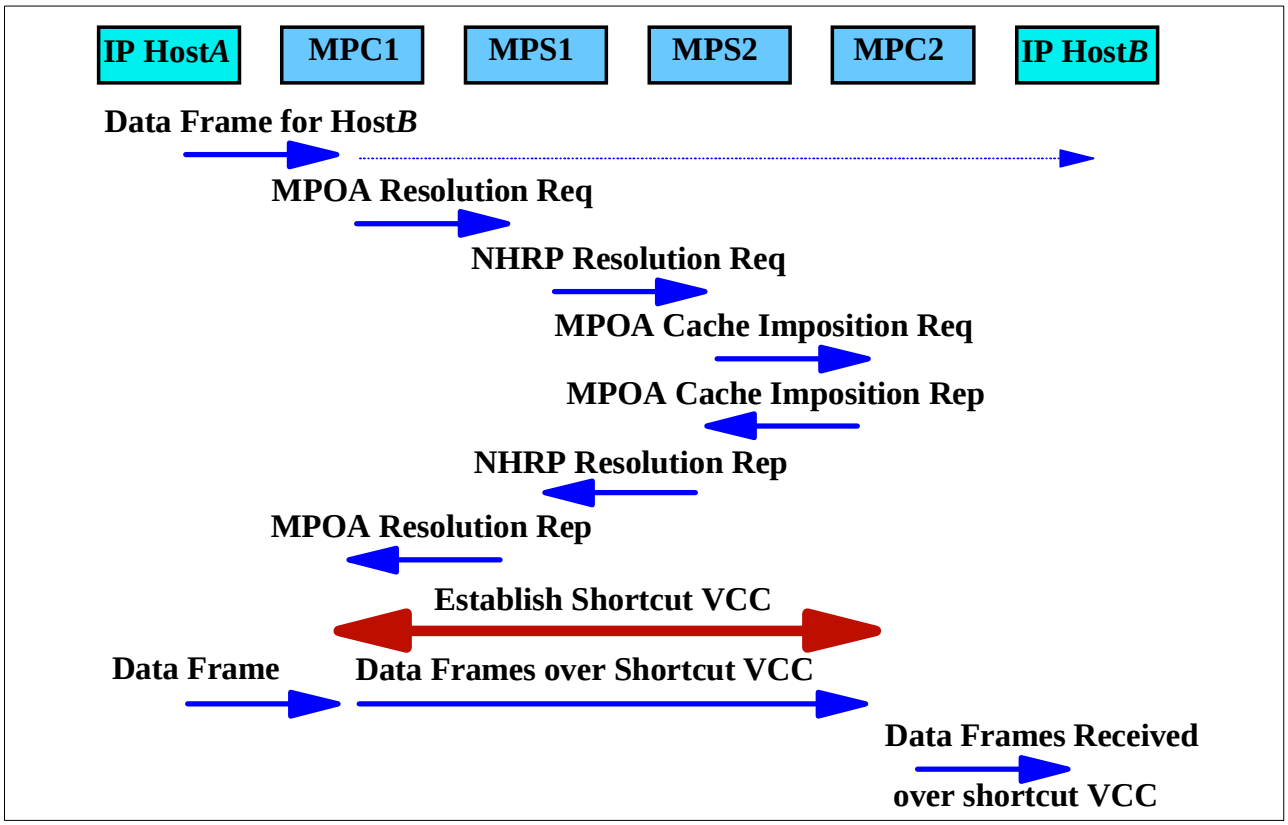


Figure 11. MPOA Shortcut VCC Establishment Flows

In Figure 11 *HostA* is transmitting data to *HostB*. At a certain moment MPC1 detects that the flow of data exceeds the threshold and a shortcut may be created. For this purpose MPC1 sends an MPOA Resolution Request for *HostB* address to MPS1. MPS1 is not serving *HostB* and thus it must convert the MPOA Resolution Request into an NHRP Resolution Request. This request is sent along the chain of MPSs until the MPS is found which serves the MPC2. MPS2 will respond to the NHRP Resolution Request with the ATM address of MPC2 because it serves MPC2 *and* because it already has an ARP cache entry for *HostB* showing the MAC/ATM addresses of the LEC associated with MPC2[5].

Before responding to the NHRP Resolution Request, MPS2 imposes an *egress cache entry* on MPC2. The cache entry is required in the egress MPOA Client so that the MPOA Client knows the data link layer (DLL) header that MPS2 would use to transmit frames to *HostB*.

[5] Remember that traffic has already been flowing along the “default path” shown in Figure 8 on page 18, which means that MPS2 has been routing this traffic and therefore has an LE-ARP cache entry for the ATM address through which traffic destined to HostB should be forwarded.

MPC2 records the cache entry and responds with an MPOA Cache Imposition Reply that includes an ATM address that can be used to set up a shortcut VCC for traffic destined to *HostB*.

After receiving the Cache Imposition Reply, MPS2 inserts the ATM address into an NHRP Resolution Reply that is sent back to MPS1. MPS1 transforms the NHRP Resolution Reply into an MPOA Resolution Reply that is sent to MPC1. MPC1 uses the ATM address to set up a shortcut VCC to MPC2. Once the shortcut VCC is established, traffic destined for *HostB* is transmitted over the shortcut VCC, bypassing the intermediate routing hops at MPS1 and MPS2. When MPC2 receives the frames over the shortcut VCC, it inserts DLL header information from the corresponding cache entry for the address of *HostB* and delivers the frames to *HostB* as if they were received via the routed path from MPS2.

If the egress MPOA Client does not respond to the Cache Imposition Request, for whatever reason, the MPOA Server in turn will not respond to the NHRP Resolution Request and no shortcut will be established. This differs from NHRP in that no action is required by the egress NHRP Client before shortcuts can be set up to it.

1.2.3 MPOA vs SuperELAN

A SuperELAN is a collection of Emulated LANs which are *bridged* together but which can establish Data Direct VCC shortcuts between clients in different ELANs without having to pass through a bridge. It is a function provided by MSS and is not a formal standard.

MPOA allows a collection of Emulated LANs which are connected with routers to achieve the same benefits of Data Direct VCC shortcuts between clients in different ELANs without requiring the traffic to pass through a router.

MPOA automatically provides the benefit of routers dividing a network into broadcast domains and restricting the scope of broadcasts to each separate routed ELAN. SuperELAN requires additional configuration using Bridging Broadcast Manager (BBCM) and Dynamic Protocol Filtering (DPF) to restrict broadcast traffic.

A complete discussion of SuperELAN can be found in *MSS Release 2.1, Including the MSS Client and Domain Client,* SG24-5231.

1.2.4 Local Shortcuts

One final thought. Many LANs which attach to ATM networks already themselves comprise multiple IP subnets. One apparent advantage of NHRP, and therefore an apparent disadvantage of MPOA, is that an Edge Router can be used to route traffic between LAN subnets. Since with MPOA there is no Edge Router, it would seem that either a separate LAN router is required, or else routed LAN traffic will travel twice across the ATM network to and from the routing function provided by the MPOA Server.

MPOA provides a nice solution for this. In the same way as for establishing shortcuts across the ATM network, the MPOA Client will establish “internal” shortcuts for this sort of traffic, which will prevent the data frames from entering the ATM network in the first place. Figure 5 on page 11 shows an example of this,

in which traffic between the devices marked “2” and “4”, which are on different IP networks, is switched internally in the MPOA Client. Naturally, the same rules are used as for ATM shortcuts, so that internal shortcuts are only established if a traffic flow is identified which exceeds the configured rate criterion.

To be absolutely accurate, the ATM Forum MPOA specification does not require that these shortcuts be set up “internally”, and notes that a simple implementation could be where the MPOA device sets up a VCC to itself. It also observes that an internal shortcut is a more efficient method of achieving the same end, and this is in fact the action that IBM MPOA Clients take.

1.3 Forum Compliance and IBM Extensions

This section describes the formal MPOA standard and why IBM has implemented additions over and above the standard.

1.3.1 “Tagged” and “Non-Tagged” Frame Formats

The ATM Forum standard AF-MPOA-0087.000 states that, by default, data frames transmitted across MPOA shortcuts are encapsulated according to the rules defined in RFC 1483[6]. It makes two further points:

1. That alternative encapsulation methods can be negotiated between MPOA Clients, but that all MPOA devices must be able to use the default encapsulation on all VCCs.
2. “MPOA Tagged Encapsulation” is defined for optional use in which a four-byte tag field follows the LLC/SNAP header.

One significance of the “tagged” format is that it reduces the overhead on the egress MPOA Client. Without the use of a tag, data frames are sent using standard RFC 1483 encapsulation, which defines several formats for IP and IPX packets. The egress MPOA Client must use the destination Layer 3 address contained in the packet as a key into its tables, where it will find the DLL header it must use on the front of the frame which it then transmits over a LAN interface. Remembering that we are talking about *Multiprotocol* over ATM, which means that the egress MPOA Client first of all needs to work out what protocol is being used before extracting the relevant Layer 3 information and then looking up the DLL header in its tables. “Tagged” format, however, is identified by the specific value x’884C’ in the “EtherType” field in the RFC 1483 frame (a value that is therefore reserved for this specific use by the ATM Forum) which tells the egress MPOA Client that it should use the four-byte tag field contained in the frame to look up the appropriate DLL header in its tables. Because the “tagged” format of RFC 1483 encapsulation does not explicitly indicate the Layer 3 protocol being used in each frame, the egress MPOA Client will use different sets of tag values for different L3 protocols.

Another reason for using the “tagged” format is that there are cases in which an egress MPOA Client will receive a *Cache Imposition Request* with the same Layer 3 destination address and the same source ATM address as an existing cache entry but with a different DLL header. Use of the tag allows the egress MPOA Client to differentiate between flows which will be received subsequently on the same shortcut VCC; without the use of a tag the egress MPOA Client either has

[6] Multiprotocol Encapsulation over ATM Adaption Layer 5, July 1993

to provide a different destination ATM address for the new cache entry or has to refuse the new Cache Imposition Request.

Even though this tag value is contained in the frame sent by the ingress MPOA Client, it has significance to the egress MPOA Client only because this is where the tag originated! When a shortcut is set up, the MPS serving the egress MPC sends the MPOA Cache Imposition Request to the egress MPC. This request contains all the information needed by the egress MPC including the DLL header, which the egress MPC stores in an internal table. If the egress MPC supports MPOA tagged encapsulation it will send back a tag value in the Cache Imposition Reply to the MPS, and it's likely that this tag value is something simple such as a direct index into the egress MPC's table of all DLL headers that it has saved. The MPS now sends this information back to the ingress MPC. The ingress MPC now understands that the shortcut will be based on tagged encapsulation and will put the tag in the frames for this shortcut. The result is that the data frames will arrive at the egress MPC with the tag, the egress MPC will read the frame, notice the ethertype x'884C' and, directly behind this field, the tag which will point to the entry in its cache.

The standard flow for the shortcut setup is shown in Figure 11 on page 24.

1.3.2 LANE Encapsulation and Vendor Extensions

Although MPOA is designed for use with ATM networks which implement LAN Emulation, the formal standard states that data frames transported over the ATM shortcuts use RFC 1483 encapsulation and that an ingress MPOA Client can only establish a shortcut to another MPOA Client. The specification does, however, define a mechanism for vendor-private extensions to the standard, and as long as all MPOA implementations comply with the basic specification there is no problem with vendors such as IBM providing extensions to the standard.

IBM has used this capability by designing an open set of extensions that support LANE encapsulation for traffic transported over MPOA shortcuts. These extensions have already been implemented as part of the NHRP function delivered in Release 1.1 of IBM's Multiprotocol Switched Services (MSS) Server, and MPOA implementations can also realise benefits from the same extensions. The value of LANE encapsulation is that it allows the ingress MPOA Client to send data to an egress device in exactly the same format as if the packet had originated from the last-hop router in the default data path. This allows shortcuts to be set up *from* MPOA Clients *to* standard LANE devices. The extensions are an important value-add because they extend the reach of MPOA shortcuts to the large installed base of LANE equipment, including LAN switches with ATM uplinks.

Note that one of the differences in implementation of LANE encapsulation when compared with the "standard" encapsulation types is that now the ingress MPOA Client has to substitute the DLL header, compared with RFC 1483-encapsulation in which the egress MPOA Client performs the substitution. This means that one of the extensions to the MPOA shortcut setup process for LANE encapsulation is a mechanism for the DLL header to be sent back to the ingress MPOA Client, which will then be stored in the ingress MPOA Client's cache.

Another advantage of LANE encapsulation arises even when the egress device *is* another MPOA Client. In the case of the MSS Client on the IBM 8270 Token Ring

Switch, all data packets received from the ATM network in this format can be switched directly to the appropriate LAN port in hardware. The 8270 also supports the RFC 1483 standards defined in the formal MPOA specification, but because it has to handle these packets in software it will always indicate a preference for receiving frames using the LANE format, ensuring that this is the format that is used wherever possible.

1.3.2.1 Some Notes on the Usage of MPOA

- Both normal RFC 1483 encapsulation and MPOA Tagged Encapsulation are part of the standard. Not supporting both encapsulation methods is against the standard. The egress MPOA Client elects which format to choose by the presence or absence of a tag value in its Cache Imposition Reply.
- Although the use of Tagged RFC 1483 Encapsulation reduces the processing load on the egress MPOA Client, it does not add to the processing load of the ingress MPOA Client by any significant amount. The ingress MPOA Client simply stores the tag value in its tables along with all the other information it requires for this particular shortcut.
- When in Figure 11 on page 24 *HostB* communicates with *HostA*, the reverse path to the original shortcut, the same procedure must be followed again to establish a shortcut in the opposite direction. However it is not necessary to establish a separate VCC: if a VCC is already established in one direction it will be used in the reverse direction once the MPOA setup process has completed for this shortcut.
- The ingress must have a certain knowledge of the encapsulation. When the egress dictates MPOA-tagged encapsulation, the ingress must be able to put the right ethertype x'884C' and the appropriate tag in the frame.
- The 8371 Multilayer Ethernet Switch in the egress position will always use MPOA-tagged encapsulation. As an ingress MPOA Client it supports both RFC 1483 methods as well as LANE encapsulation.
- The 8270 prefers LANE encapsulation for performance reasons, but supports both non-tagged and MPOA-tagged as well, as dictated by the standard.

1.3.3 ATM Addresses and MAC Addresses: MPOA and LANE Client

The ATM Forum MPOA specification[7] contains considerable discussion on the relationship between ATM addresses and MAC addresses for the LEC and MPC/MPS. In summary:

- Each LEC for which MPOA is enabled is associated with exactly one MPC or one MPS.
- A single LEC may have multiple MAC addresses.
- A single LEC with multiple MAC addresses may choose to have MPOA enabled for a subset of these MAC addresses.
- If an MPOA device has a LEC with a set of MAC addresses which are associated with an MPC as well as a set of MAC addresses which are not associated with any MPOA capability, then separate LEC ATM addresses must be used for each set of addresses. This is because other MPOA devices learn MPOA capabilities from LANE data frames by relating MAC addresses to known MPOA-capable LEC ATM addresses.

[7] Multi-Protocol Over ATM Version 1.0, AF-MPOA-0098.000, July 1997

- Each MPC or MPS serves a set of LECs and has a single Control ATM Address.
- The Control ATM Address can be the same as an ATM address used by one or more of its LECs.
- If there are multiple MPCs in a device then each MPC must serve a disjoint set of LECs and each MPC must use a different MPC Control ATM Address (this applies to the 8371 Multilayer Ethernet Switch).
- The MPC supplies each LEC for which it is enabled with the MPC's Control ATM Address (via the MPOA Device Type TLV). This value is then used by the LEC in every LE-ARP response it sends, therefore indicating to the recipient that there is an MPC serving the LEC and providing the actual Control ATM Address of the MPC. The same TLV is also sent as part of the MPOA Registration Request when the MPOA Client registers with the MPOA Server, and this enables the LES to forward the MPC's Control ATM Address when it responds to LE-ARP requests.

1.4 Communication Examples

Several possibilities for communication across an ATM network exist. This table gives examples of these possibilities:

Table 2. Communication Possibilities for Different Client Combinations

Communication Partners	Method of Communication
Single subnet of a routable protocol in a single ELAN	LANE Data Direct VCC only one LAN type (token-ring or Ethernet) concurrently supported
Multiple subnets of routable protocols in a single ELAN	LANE Data Direct VCC, only one LAN type concurrently supported
Single subnet of a non-routable protocol in a single ELAN	LAN Data Direct VCC, only one LAN type concurrently supported
Multiple subnets of non-routable protocols in a single ELAN	LAN Data Direct VCC, only one LAN type concurrently supported
Multiple subnets of routable protocols in more than one ELAN	External routing between ELANs. This can be done by the MSS. This is based on the ATM standards. A routed configuration allows for multiple LAN types. With the MSS a SuperELAN can be made. The SuperELAN supports Data Direct VCCs between ELANs, and only one LAN type within a SuperELAN.
Multiple subnets of non-routable protocols in more than one ELAN	External bridging between ELANs. This can be done by the MSS. The MSS supports multiple bridging options. Based on the bridge option multiple LAN types can be used concurrently.
CIP Client and ELAN Client	ATM standards provide no direct communication possibility between the two. An external router supporting both protocols must be used.

Communication Partners	Method of Communication
NHRP Clients	Standard NHRP Clients use Classical IP over ATM and use NHRP for address resolution across subnets. After the resolution a VCC can be established. IBM's extensions make NHRP usable for the LAN Emulation environment as well.
MPOA Clients in a single ELAN	LANE Data Direct VCC
MPOA Clients in multiple ELANs	Initiall communication through the MPOA Server. When threshold is exceeded; shortcut VCC between MPOA Clients
MPOA Clients and CIP Clients	Standards-based communication must take place through an external router. This router must support LANE and CIP. IBM's MPOA extensions make it possible to create the shortcut.
MPOA Clients and NHRP Clients	Standards-based communication takes place with an external router. IBM's extensions make it possible to create the shortcut VCC.
MPOA Clients and LANE Clients	IBM Extensions allow MPOA Clients to set up shortcuts to LANE Clients; LANE Clients must send traffic to a router in ATM network.

1.5 Conclusion

Both Layer 3 switch networks and MPOA networks provide a network infrastructure with much higher capacity than existing networks, and each of these types of network will have its own marketplace. The current Layer 3 switch networks will not be as big as the MPOA networks; a typical Layer 3 switch network will not normally be bigger than a single building. Other routing functions are needed which cannot be provided with a Layer 3 switch. Connections between buildings will use with traditional routers with some kind of wide area network protocol such as frame relay or even ATM.

An MPOA network can become much larger than a single building. Using ATM both as a local area network protocol and as the wide area network protocol creates a network that will present a single image to the user and management system. Sessions that are established in one building can be switched across the entire network without the need of intermediate routers.

One of the key differentiators of ATM were the several choices available in the speed of the connections. ATM in local area network now has speeds between 25 Mbps and 2.4 Gbps and Ethernet has speeds between 10 Mbps and 1000 Mbps. ATM will be the technology of choice when used in a larger mixed environment with multiple sites connected with ATM wide area network connections. The backbone that is built must be capable of transporting current and future multi-gigabit traffic volumes over an available, congestion free and low delay path. This infrastructure must be dynamic and easy to upgrade to future needs.

Chapter 2. Layer 3 Switching Implementations

This chapter looks at ATM networks and discusses IBM's implementation of Layer 3 Switching using Multiprotocol Switched Services (MSS). It positions the MSS Server, MSS Client and MPOA Client in the 8371 Multilayer Ethernet Switch in relation to Layer 3 Switching.

2.1 Summary

Layer 3 Switching is the up and coming network technology. With an ATM network as the core of the network it has become possible to build a network that is scalable, manageable and expandable.

The MSS Server is the cornerstone of IBM's ATM network strategy. At the edges of an ATM network, MSS clients are used to enable Layer 3 Switching. The major aspects of these services are covered in this chapter, without going into the implementation and configuration details which are covered in the second half of the book.

2.2 MSS, ATM and Layer 3 Switching

This section discusses in detail the MSS and its role in the network: what it is, when and why it should be used. Examples of realistic network designs are used to illustrate.

The products which implement MSS are described in some detail. No description of the ATM switch is included. Without the implementation of functions provided by MSS, only devices which can use ATM "natively" can attach to an ATM switch.

2.2.1 MSS Server

The MSS Server connects to any ATM switch and is designed to support devices that run on ATM, including ATM-attached workstations, ATM bridges, and ATM LAN switches. In this environment, the MSS Server provides these basic functions:

1. ATM Forum standard LAN Emulation
2. IETF standard Classical IP over ATM
3. Routing and bridging support

Without the MSS Server, only a rudimentary subset of these functions is implemented by the ATM Control Point Switch.

The MSS Server combines all the server functions which are needed to make ATM usable in a real network environment.

There are two physical types of MSS Server: the IBM 8210 Nways Multiprotocol Switched Services (MSS) Server, which is a stand-alone product, and the IBM Multiprotocol Switched Services (MSS) Server module, which is installed as a module in the IBM 8260 Nways Multiprotocol Switching Hub (8260) or the IBM 8265 Nways ATM Switch (8265). The stand-alone version is connected to the ATM network over 155 Mbps optical fibre cable equipped with industry standard SC connectors. The MSS Server Module connects to the ATM network when it is installed in the 8260 or the 8265. Several models exist of both the stand-alone

and the blade version. The evolution of the 8210 and its software is explained in 4.2, “MSS Family History (Software)” on page 74.

The MSS Server enables you to move from a routed network to a switched network. Specifically, it provides functions required to build a data network in an ATM environment.

The MSS Server provides all the LAN emulation server functions as well as the server function for Classical IP over ATM. Its bridging and routing functions can connect these virtual LANs as if they were physical LANs, IP subnetworks, or IPX subnetworks. The MSS Server also offers two important server functions:

1. Multiprotocol over ATM (MPOA) server support. It supports both token-ring and Ethernet MPOA clients.
2. Next Hop Resolution Protocol (NHRP) Server support.

2.2.1.1 ATM Forum-Compliant LAN Emulation

ATM Forum LAN Emulation (LANE) allows ATM networks to appear as LANs to provide a migration path to ATM to protect the investment in current LAN hardware and software.

By providing LAN Emulation, the MSS Server offers the opportunity to connect ATM devices with Ethernet or token-ring devices, supporting ATM backbones and gradual migration while protecting the investment in LAN hardware and applications.

Emulated LANs are not based on physical topology, like existing shared media LANs, but are logical groupings of endstations. Having stations logically grouped allows greater flexibility in handling moves, additions, and changes to the endstations.

The server implements the three basic functions required to run a LAN Emulation network, which are required to enable emulation of a “broadcast” LAN environment over a “non-broadcast” switched ATM infrastructure.

The *LAN Emulation Server* (LES) implements the control coordination for the emulated LAN. The LES provides a facility for registering and resolving MAC addresses and/or descriptors to ATM addresses. Clients register the MAC addresses they are emulating with the LES and they also query the LES when the client wishes to resolve a MAC address and/or route descriptor to an ATM address. The LES will either respond directly to the client or forward the query to other clients so they may respond

The *Broadcast and Unknown Server* (BUS) handles data sent by an LE client to the broadcast MAC address (’FFFFFFFFFFFF’). All multicast and initial unicast frames that are sent by a LAN Emulation Client before a data direct VCC has been established are handled by the BUS. The BUS must always exist in an emulated LAN and all LAN Emulation Clients must join its distribution group.

An ATM network can consist of several emulated LANs. The *LAN Emulation Configuration Server* (LECS) assigns LAN Emulation Clients to an emulated LAN based on its configuration database, its own policies and the information it receives from the respective LE Clients. It assigns any client that requests configuration information to a particular emulated LAN service by giving the client the LES’s ATM address. This method supports the ability to assign a client to an

emulated LAN based on either physical location (ATM address) or the identity of a LAN destination that it is representing. The LECS is an optional component of an ATM LANE network, but it is extremely useful because it is normally configured with a standard “well-known” ATM address which the majority of LAN Emulation Clients are pre-configured with; using LECS it is possible to attach an ATM edge device to an ATM network without any pre-configuration necessary on the edge device itself[1].

2.2.1.2 MPOA Support

The MSS Server’s MPOA support is fully compliant with the ATM Forum MPOA specifications. MPOA allows Layer 3 protocols such as IP and IPX to establish inter-subnet shortcuts for efficient use of resources in an ATM network. MPOA works in the LAN Emulation over ATM environment specified by the ATM Forum and uses the IETF’s NHRP standard (RFC 2332) for inter-server communications.

MPOA is fully integrated into the MSS Server. All MSS advanced functions are available to MPOA users. MSS advanced functions allow you to build reliable and scalable ATM networks by implementing additional features such as Broadcast Manager, Super VLANs, NHRP, route clients, Dynamic Protocol Filtering, ARP Servers, LES and BUS redundancy, and IP Gateway redundancy.

The MSS Server support for Next Hop Resolution Protocol complies with the standards. The NHRP Server (NHS) is implemented in the MSS Server and works with the NHRP Clients (NHCs) at the edge of the network. The MSS Server provides support for zero-hop routing for ATM-attached endstations if these stations have NHCs; when no NHC is installed it will perform one hop routing.

2.2.2 IBM 8270 Nways LAN Switch

The IBM 8270 Nways LAN Switch and the 8371 Multilayer Ethernet Switch are examples of systems which provide the function of an edge device when used in an ATM network. Both the 8270 and the 8371 have MPOA Client functions as described in 2.2.3, “MSS Client” on page 36. The 8270 provides full MSS Client functions and the 8371 has LAN Emulation and MPOA Client functions.

[1] For example, an edge Ethernet switch can use the LECS to assign itself to the first Ethernet ELAN defined in the ATM network. If there is more than one Ethernet ELAN in the network it would probably be more usual to configure the edge device with the name of the ELAN it is to join, but this is probably the only parameter which requires configuration. LECS dispenses with the need to configure ATM addresses in the edge device at all.

Figure 12. IBM 8270 Nways LAN Switch Model 800

The 8270 is a chassis-based token-ring switch. Two models are currently on the market. The model 600 has six feature slots and the model 800 has eight. A second power supply can be added to the model 800. Both systems use the same universal feature cards:

- Four-port token-ring enhanced unshielded/shielded twisted pair (UTP/STP)
- Two-port token-ring enhanced Fibre
- One-port ATM II UFC
- MSS Domain Client
- MSS Client Multimode Fibre (MMF)
- MSS Client Singlemode Fibre (SMF)

The first two cards act as token-ring ports in a hub. The ATM II UFC single physical port provides the following features:

- 155-Mbps data rate (OC3)
- Multimode fibre (MMF) subscriber connector (SC)
- Constant bit rate (CBR), variable bit rate (VBR), and unspecified bit rate (UBR) virtual channel connection (VCC) types
- 3072 VCCs
- ATM user-network interface (UNI) Specification, Versions 3.0 and 3.1

The single physical port allows multiplexing of data from multiple logical ports. An ATM UFC logical port is configured as a LAN Emulation Client (LEC). On one ATM UFC, up to 32 LECs can be configured, with a maximum of eight LECs enabled at one time.

The ATM UFC implements the industry-standard ATM Forum LAN Emulation over ATM: Version 1.0 specification. ATM LANE allows for the seamless flow of token-ring and Ethernet data over an ATM network, independent of higher-layer protocols. The UFC can be placed in several switches:

- 8270 Nways Token-Ring LAN Switch model 800
- 8270 Nways Token-Ring LAN Switch model 600
- 8271 Nways Ethernet LAN Switch model 108
- 8271 Nways Ethernet LAN Switch model 216
- 8271 LAN Switch module
- 8272 Nways Token-Ring LAN Switch Model 108
- 8272 Nways Token-Ring LAN Switch Model 216
- 8272 LAN Switch Module

All the ATM UFCs in these switches will automatically join an ELAN without any pre-configuration necessary provided that an ELAN of the right type is defined in the ATM network and that the ATM network has implemented LECS with the standard “well-known” ATM address.

Figure 13. IBM 8270 Nways LAN Switch Model 600

The ATM UFC can be used in two phases of network migration:

1. The first phase is the implementation of an ATM backbone. An ATM backbone is generally used to upgrade the capacity of the backbone without making changes to the rest of the network. Using the ATM UFC in conjunction with the switch will increase the speed of the backbone. The users might experience another performance gain since switches have in general a better throughput than bridges.
2. In the second phase high speed and frequently-used servers are directly connected to ATM to take advantage of ATM’s scalability. By using LAN Emulation, the ATM-based workstations are able to use existing legacy LAN applications with minimal or no upgrade cost for moving to ATM.

A network with a high capacity backbone might not create the immediate relief that might be expected. Servers often have the same adapters that users have, so it is questionable whether a server with a regular LAN adapter would be capable of handling all the user traffic that the enhanced backbone now directs to it. In a second phase of a network migration it would make sense to attach the

server directly to ATM and to create a high-speed access to such a server. This server should support LAN Emulation, and now both the connection between the server and the users and the ability of the server itself to process the additional traffic is now upgraded.

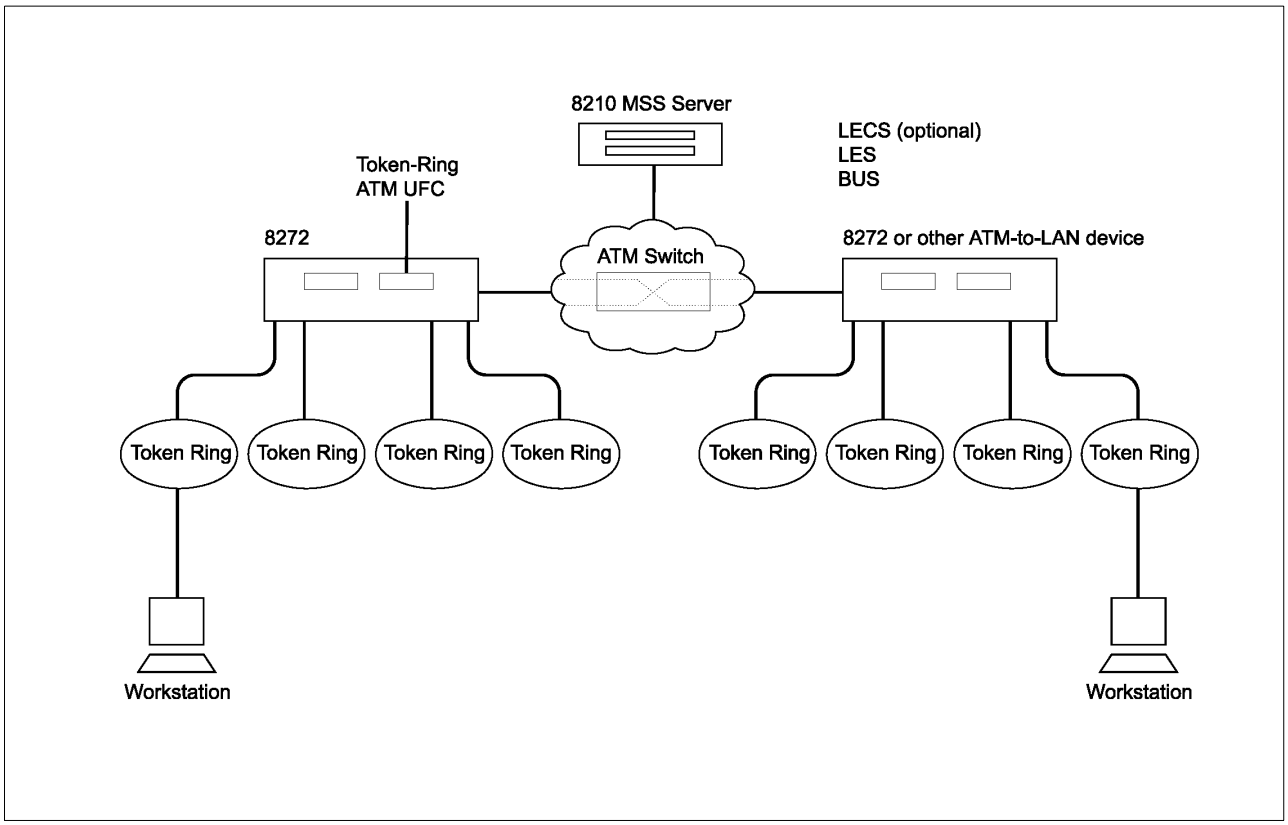


Figure 14. Typical Usage of the ATM UFC

2.2.3 MSS Client

The *Multiprotocol Switched Services Client* (MSS Client) and the *Multiprotocol Switched Services Domain Client* (MSS Domain Client) are collectively known as the *MSS Family Client*. The MSS Domain Client is used for enhancing the bridge function of the switches. The MSS Client additionally provides an ATM attachment and provides MPOA Client and NHRP Client support.

The MSS Client UFC is designed for installation in the following IBM token-ring Nways LAN switches:

- 8270 Token-Ring LAN Switch Model 800
- 8270 Token-Ring LAN Switch Model 600
- 8272 Token-Ring LAN Switch Module for the Nways 8260 Multiprotocol Switching HUB.

For the remainder of this book we will only discuss the MSS Client UFC in the context of the 8270 Token-Ring LAN switch. More information on the original implementation of the MSS Client and MSS Domain Client can be found in *MSS Release 2.1, Including the MSS Client and Domain Client*, SG24-5231.

Table 3 gives an overview of the functions:

Table 3. Interfaces, Protocols, and Services Supported by MSS Family Client

Feature	MSS Client	MSS Domain Client
Interfaces		
Token-Ring LEC	yes	no
Ethernet LEC	yes	no
Token-Ring proxy LEC	yes	no
Internal Token-Ring interface	yes	yes
Internal Ethernet interface	yes	yes
FastTR over ATM	yes	no
Protocols and Features		
Classical IP	yes	no
IP	yes	yes
Banyan VINES	yes	yes
AppleTalk	yes	yes
IPX	yes	yes
Source-Route Bridging	yes, token-ring only	yes, token-ring only
NHRP	yes	no
LAN Network Manager	yes	yes
MPOA	yes	no
PVLAN	yes, token-ring only	yes, token-ring only
CIP ARP server redundancy	yes	no
QoS LEC	yes	no
MARS Client	yes	no
OSPF/MOSPF	yes	yes
RIP	yes	yes
RIP2	yes	yes
DVMRP	yes	yes
BGP	yes	yes
Transparent Bridge	no	no

For completeness, we should mention FasTR, which allows two ATM adapters to be connected over a point-to-point link without the need for an ATM switch and without the need for defining ATM over the interfaces. Although ATM AAL5 cells are actually transmitted over the link, the adapters convert token-ring frames to- and from AAL5 at each end, and therefore provide the appearance of a fast token-ring connection. A single FasTR connection provides approximately 110 Mbps full-duplex bandwidth between the two adapters. Currently the MSS Client

installed in a token-ring switch and the 2216 support this feature, so one use for FasTR would be to provide high-speed mainframe access when connecting to a channel-attached 2216. FasTR is a Layer 2 transport mechanism; the other method of directly connecting ATM adapters without using an intervening ATM switch would be to define RFC 1577 (Classical IP) over the link, but this would then only support IP traffic and require IP routing engines at each end of the link.

When the source-route bridging (SRB) function of the MSS Family Client is enabled, the Client processor handles the Spanning Tree Protocol and all the forwarding of the explorer frames. The SRB function in the base switch continues to process all known and unknown specifically routed frames (SRFs) for unicast and multicast traffic. This processing creates a distributed system environment between the base switch and the MSS Family Clients which enhances performance. The MSS Family Client takes over some of the SRB function from the token-ring switch and offers higher performance as a result.

The MSS Client provides an ATM interface with support for LAN Emulation, MPOA, NHRP, and native protocol support for IP and IPX.The MSS Client provides a single OC3 ATM interface for connecting to the campus backbone. When the MSS Client SRB function is enabled, the traffic between the ATM interface and the switch domain is bridged through hardware. When the traffic between the ATM interface and the switch domains is routed, the software path of the MSS Client's CPU is used.

MSS Client and SRB

To implement MPOA on a token-ring edge switch, the MSS Client has to be configured for source-route bridging between the LAN ports of the switch and the ATM interface of the MSS Client. This is because the MPOA Client code interfaces with the SRB code; MPOA gets a chance to look at each frame as it passes through the bridge and has to decide whether or not to forward this frame over an MPOA shortcut.

2.2.3.1 Bridging

The MSS Family Client does not support bridging on an Ethernet LAN switch. To enable the enhanced source-route bridging (SRB) support, you must configure SRB support on the MSS Family Client after installing the MSS Family Client hardware in the base LAN switch. You must explicitly configure SRB support on the MSS Family Client. When you enable this support, the MSS Family Client will reconfigure the base SRB support in the LAN switch to distribute the workload between the base switch and the Family Client. You can still view the SRB configuration and statistical information on the base LAN switch, but you will not be able to change configuration information for SRB on the LAN switch. When the MSS Family Client becomes disabled, the base LAN switch SRB configuration is restored and the system administrator can make configuration changes to the SRB feature on the LAN switch.

One significant difference between the MSS Client's UFC ATM support and the ATM UFC is that you can place the ATM UFC in the same domain as legacy LAN switch ports. This cannot be done with the MSS Client's ATM support. The SRB feature in the MSS Client must be used to bridge between LAN switch domains and the MSS Client's ATM interface. This applies even when MPOA is not implemented.

2.2.3.2 Routing

The routing support must be explicitly configured for the MSS Family Client. When a routing interface is defined, the MSS Family Client attaches a virtual routing interface to a domain in the base LAN switch. The MSS Family Client supports routing for IP, IPX, AppleTalk and Banyan VINES, and routing support is required in order to implement the NHRP Client function.

2.2.3.3 Multiple MSS Family Clients

More than one MSS Family Client can be installed in a single LAN switch to increase overall throughput. Although multiple MSS Family Clients can reside in the same LAN switch, you can only configure one for bridging because of the way that the bridging support interacts with the bridging function of the base LAN switch. The base LAN switch is capable of the following MSS Family Client combinations, with some restrictions on which slots they can be placed in:

- One MSS Client, one MSS Domain Client
- Two MSS Domain Clients
- Two MSS Clients

However, you can install only one MSS Client or one MSS Domain Client if you have already installed an ATM UFC in the LAN switch. You cannot install any MSS Family Clients if two ATM UFCs are in the LAN switch.

Installing two MSS Family Clients allows the system to distribute routing between two processors, which provides performance benefits similar to those gained by distributing the SRB bridging function between the base token-ring LAN switch and the MSS Family Clients. The system can also use the two clients to separate the routing and bridging functions performed by the MSS Family Clients.

The following configurations are valid if used in an ATM environment:

1. Two MSS Clients can be installed in a single 8270, but only one MSS Client can be used as an MPOA Client because only one MSS Client can implement SRB and the SRB function is needed by MPOA.
2. Two MSS Clients can be active in the same 8270. Only one Client can be used for bridging. The other client can be used for routing. Both clients can be used concurrently for routing (they will appear as two separate router instances with different addresses).
3. For redundancy purposes the only way to have two active MPOA Clients attached to a segment is to use two 8270s. The source path taken by the routed traffic will be determined during source route discovery and the MPOA Client in that source route path will be the one that is used.

When the MSS Client is configured for the SRB functions it will modify the base 8270 SRB configuration. In the case of an MSS Client failure the original base configuration is restored. This means that if we want the base to take back SRB function in the case of an MSS Client failure, we must configure this prior to installing the MSS Client. Failure to do so will result in a configuration which does not perform any bridging.

2.2.4 MSS Domain Client

The MSS Domain Client performs a subset of the functions of the MSS Client, primarily because it has no ATM interface. The functions are listed in Table 3 on

page 37. Its main functions are to provide enhanced bridging support on the base switch and to provide routing support.

The MSS Domain Client Universal Feature Card is designed for installation in the following IBM Nways LAN switches:

- 8270 Token-Ring LAN Switch model 800
- 8270 Token-Ring LAN Switch model 600
- 8271 Ethernet LAN Switch Module for the Nways 8260 Multiprotocol Switching Hub
- 8272 Token-Ring LAN Switch Module for the Nways 8260 Multiprotocol Switching Hub

MSS Release 2.1, Including the MSS Client and Domain Client, SG24-5231 is recommended for further reading about the MSS Domain Client.

2.2.5 8371 Multilayer Ethernet Switch

The 8371 Multilayer Ethernet Switch is a new Ethernet switch from IBM. The system comes standard with 16 10/100 Base-TX auto-sensing Fast Ethernet ports for UTP or STP cabling. It comes in two models. The 8371 Model A16 is a stand-alone system with two feature slots. The other model, the 8371 Multilayer Ethernet Switch (MLS) Module, is a module for the 8265 ATM switch and has a direct ATM attachment to the ATM backplane of the 8265. The module is a native 8265 module. Because of the direct ATM connection the module has only one feature slot which cannot be used for the ATM feature card. The 8371 Multilayer Ethernet Switch (MLS) Module comes either with 16 10/100 Base-TX ports or 16 100 Base-FX ports. The module provides an OC-12 connection to the ATM backplane of the switch and makes full use of the redundant features of the 8265 and the ATM network such as:

- Load balancing via PNNI
- Resilient uplinks
- PNNI rerouting
- Fault tolerant chassis

8265 Code Requirements

An 8265 and CP/SW with operational code 4.1 are needed to operate the 16-port 10/100 Base-TX and 16-port 100 Base-FX versions of the 8371 module

8260 Version

There is no version of the 8371 Multilayer Ethernet Switch implemented on a module for the 8260 ATM switch.

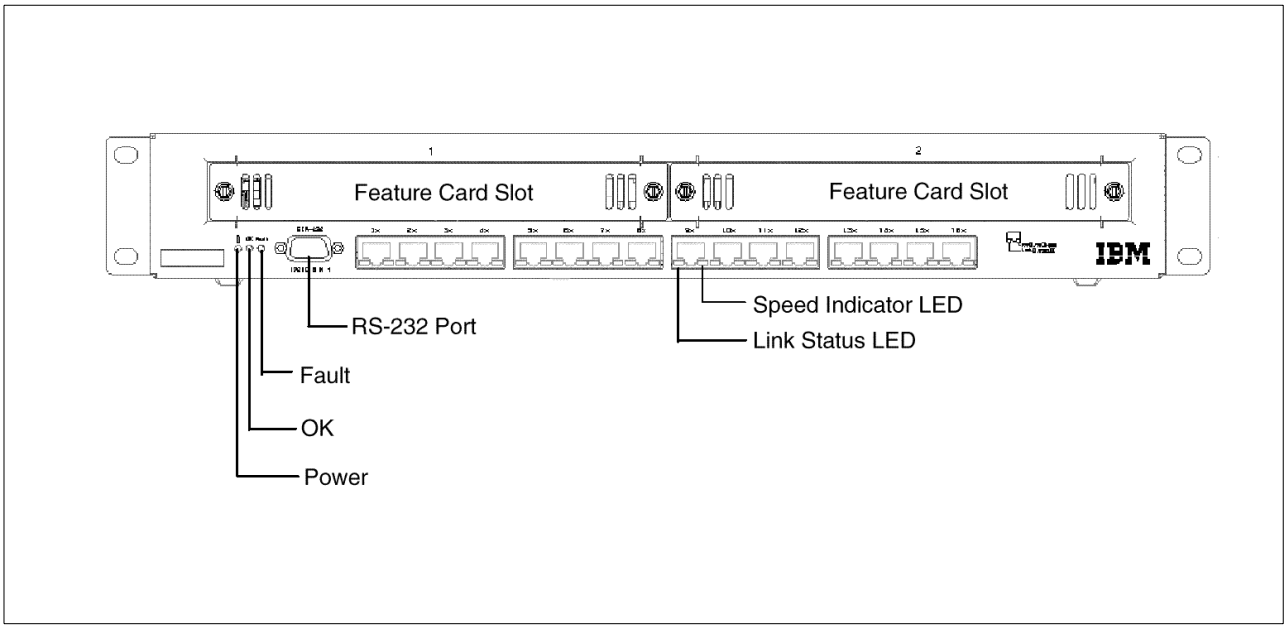


Figure 15. 8371 Multilayer Ethernet Switch Model A16 Front View

Currently three feature cards are available:

- Two-port 155-Mbps ATM MMF expansion card
- Eight-port 10/100 Base-TX expansion card
- Eight-port 100 Base-FX MMF expansion card

IBM intends to further enhance the 8371 Multilayer Ethernet Switch with additional feature cards and software functions.

Both the 10/100 Base-TX and the 100 Base-FX cards expand the number of Ethernet ports. Both cards can be used in any configuration, leading to a maximum number of 32 10/100 ports or a maximum of 24 10/100 ports in conjunction with an ATM interface card. The ATM module can be placed in either of the two slots, but two modules cannot be installed in both slots simultaneously; remember that each module provides *two* ATM interfaces. For ease of configuration it is recommended that the card be placed in the first slot.

Number of ATM Interfaces

Note that each ATM UFC provides *two* ATM interfaces; if both of these are planned to be used simultaneously then some thought must be given to the implications of Spanning Tree on the design of the network: in some configurations one of the interfaces will end up being blocked as a result; this may or may not be the desired intent.

Figure 16. 8371 Multilayer Ethernet Switch Model A16

In conjunction with the MSS Server in an ATM network, the 8371 provides local, wirespeed IP and IPX routing between Fast Ethernet ports on the same module and MPOA one hop routing over the ATM network.

Figure 17. 8371 Multilayer Ethernet Switch (MLS) Module

The 8371 has built-in control software that provides a set of full function Ethernet bridging and routing, often using hardware for higher performance. The control software is stored in EPROM and flash memory and is automatically invoked when the switch is powered-on. The software is a development of the existing “common code” software which will already be familiar to users of products such as MSS or the 2210/2216 range of routers. The software functions include:

- MPOA support for IP and IPX routing
- MPOA support for shortcuts with other MPOA Clients (Ethernet and token-ring)
- MPOA support for load balancing/redundancy with dual ATM uplinks
- SNMP
- IP Multicast VLANs with media speed forwarding
- Self-learning IP switching, although this function is mutually exclusive with MPOA and primarily designed for an implementation of the 8371 without ATM interfaces.

The 8371 implements MPOA Client functions in its base operating code and does not require a separate MSS Client UFC. The 8371 implements a subset of the functions provided in the MSS Client UFC for the 8270.

The 8371 supports ATM QoS, which is part of the LANE Version 2 specification and allows LAN Emulation Clients to work in conjunction with the LAN Emulation Configuration Server (LECS) to attempt to set up LAN Emulation Data Direct VCCs using specific ATM Quality of Service parameters. The key attributes and benefits can be described as follows:

- An LE Client makes use of configured QoS parameters for Data-Direct VCCs.
- QoS parameters can be configured for:
  - LE Client
  - ATM Interface
- The set of QoS parameters configured are for use with ATM Forum UNI 3.0/3.1 signalling. The parameters include desired Peak Cell Rate, Sustained Cell Rate, QoS Class and Maximum Burst Size.
- Maximum Reserved Bandwidth per VCC can be configured to protect an LE Client from accepting/establishing VCCs whose traffic parameters cannot be met.
- The QoS Negotiating mechanism enables the participating LE Clients to be aware of each other's QoS parameters. A Data Direct VCC is set up using the negotiated parameters.

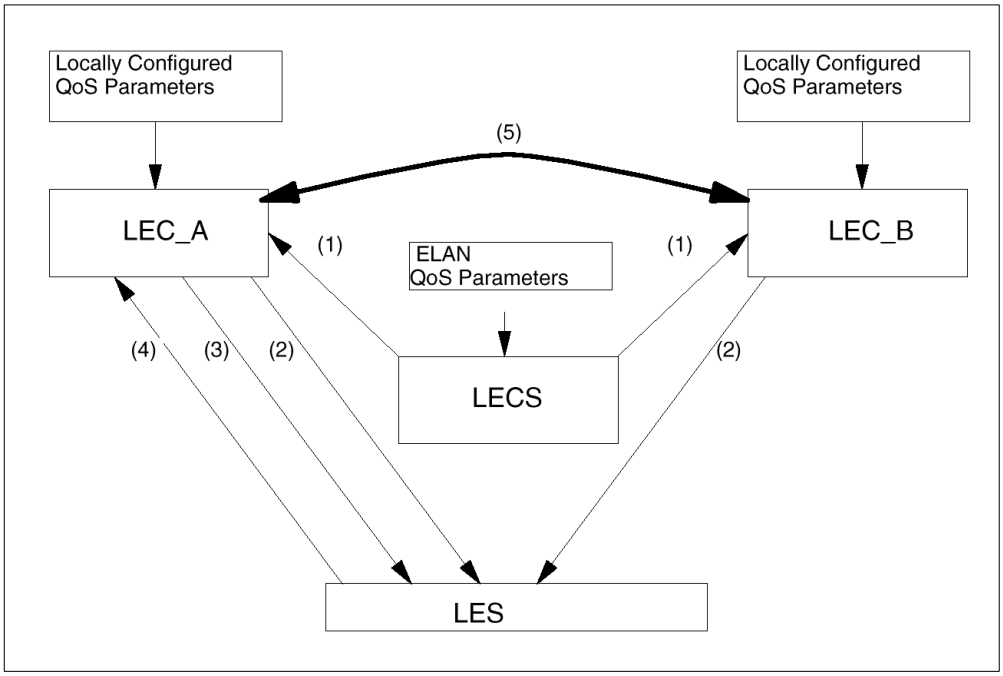


Figure 18. ATM QoS

1. When LEC_A sends an LE_CONFIGURE_REQUEST to the LECS, it will contain TLVs signifying that it is a QoS capable device. If the LECS is QoS capable it will return in the LE_CONFIGURE_RESPONSE any QoS parameters that have been defined. Any values returned by the LECS will override values configured on LEC_A

2. When a QoS-capable LEC sends an LE_JOIN_REQUEST to its LES, it will include QoS TLVs. The LES will cache these values along with the joining LEC's ATM and MAC addresses.
3. LEC_A sends an LE_ARP_REQUEST to the LES for LEC_B's ATM address.
4. The LES returns LEC_B's QoS TLVs as well as LEC_B's ATM address in the LE_ARP_RESPONSE.
5. LEC_A will compare its own QoS values with those of LEC_B, determine the best match, and then initiate a Reserved Bandwidth VCC.

The parameters used for the creation of a QoS Data Direct VCC are explained in Appendix B, "8371 LEC QoS Parameters" on page 215.

Chapter 3. Layer 3 Switching Solutions

This chapter describes some examples of MPOA implementation. The chapter aims to give the reader examples of how MPOA can be used in real networks with a mix of protocols (IP, IPX), user networks (token-ring, Ethernet) and a requirement for high availability.

The intention of this chapter is to give a high-level overview of the sorts of networks that can implement Layer 3 Switching using ATM and the latest versions of software and hardware.

There are three solutions described:

1. An example of using MPOA for IP and IPX when more than one IP/IPX subnet has to be served by a single MPC. Two important topics are covered:
    1. Dynamic protocol filtering (DPF) is used in the MSS Client to reduce broadcast, when the configuration is changed from routing to bridging
    2. Local shortcuts are explained. They make it possible to switch L3 packets between the two IP/IPX subnets that are served by the MPC.
2. Shows how the simple MPOA network of solution 1 can be extended to provide resilience against network component failure. MPC and MPS redundancy is covered.
3. Gives an example of how MPOA can be used in an environment with mixed token-ring and Ethernet user networks. Use of the 8371 Multilayer Ethernet Switch is demonstrated and explained.

3.1 MPOA for IP and IPX

In this example, we will build a basic MPOA network. We want to show the way MPOA for IP and IPX works and when to apply it. Special attention is paid to networks using several IP/IPX subnets that were previously connected with LAN routers.

In this type of network a very interesting feature of IBM MPOA Clients called Local Shortcuts will be used. This special type of shortcut allows the 8270 to switch IP/IPX frames internally between different IP/IPX subnets on the LAN ports, even including different subnets on the same LAN port.

Also, by using Dynamic Protocol Filtering (DPF), the bridge ports in the 8270 are partitioned into domains which are each assigned to an IP/IPX VLAN to keep broadcast domains within each VLAN. As its name implies, DPF is applied dynamically.

3.1.1 Introduction

It is very common that networks have more than one IP subnet connected to the same MPOA Client, such as the one depicted in Figure 19 on page 46 in which IP subnets 8.8.1.x and 8.8.4.x both connect to LAN ports of the same 8270 switch. In this case, since MPOA requires the MPOA Client to be associated with bridging function, the MSS Client cannot act directly as a router between these two IP subnets.

This seems to present itself as a disadvantage of implementing the MPOA Client in the edge switch when compared with implementing NHRP, since implementing the NHRP Client requires the MSS Client in the 8270 to be configured with routing function, which allows it also to route between different LAN networks directly - the model of an *edge router*.

As opposed to this edge router model, MPOA requires the implementation of a source-routing bridge in the MPOA Client between the ATM ELAN and the legacy segment. This means that there are two possible solutions:

1. Maintain the IP subnetting and put two IP addresses on the router in the ATM network, so that this is the entity that does the routing. This is the approach shown in Figure 19 on page 46. The MSS Server then behaves as one-armed router for this ELAN.
2. *Flatten* the IP network so that the IP mask covers both IP subnets and there is no longer any need to perform routing.

Normally the first solution is preferred since it does not require the reconfiguration of all stations, but it implies the need for all routed frames to go into the ATM cloud, to the MSS Server, and then come back again to the MSS Client.

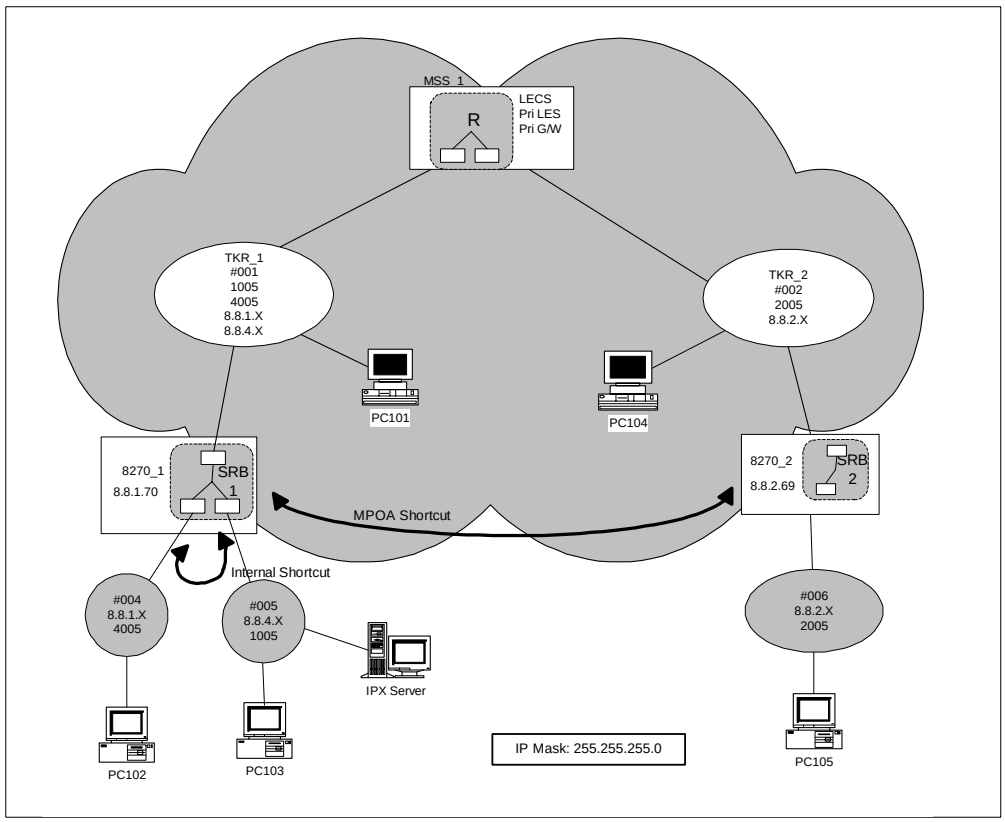


Figure 19. Solution 1: Basic MPOA Network with Several IP Subnets in Each MPC

On the other hand, both solutions have a major drawback. The MSS Client is bridging between both token-ring segments, and therefore the broadcasts in one LAN are passing to the other. Moreover, these broadcasts are being bridged into the ATM ELAN as well. Again, compare this approach with the edge router model

in which each token-ring segment is isolated from the other and the MSS Client routes between them.

In small networks this may not be a problem but some big networks require that the broadcast domains are kept as small as possible. If that is the case, the alternatives are:

1. Use a different 8270-MSS Client for each IP subnet segment, which obviously increases the cost of the solution. Although this solution appears to bring the benefit of allowing MPOA shortcuts between the two 8270s, we shall see that this is, in fact, unnecessary.
2. Do not use MPOA, and go back to the edge router model using NHRP.
3. Use DPF VLANs to keep broadcasts local to their LAN. If IP protocol is being used, then the solution would be to use IP VLANs. If it is IPX, then use IPX VLANs.
4. Use bridging broadcast manager (BBCM) to convert some broadcast frames into multicast frames, and therefore reduce the amount of broadcast in the network.
5. Use bridge filters to prevent some of the frames from being bridged to specific domains. For example, to keep NetBIOS traffic local within each Token-Ring segment.

In this example we will present an implementation of alternative 3, although alternative 4 could additionally be coupled together with alternative 3. The advantage of alternative 3 versus alternative 5 is that #3 is dynamic, thus allowing users to move stations from one token-ring segment to the other and still retain connectivity, without the need to reconfigure IP addresses.

In our network, the combination of MPOA and DPF gives the best solution because it increases the performance of the network at lower cost than the other solutions.

Using DPF VLANs, the broadcast traffic is kept within its own LAN or ELAN. The MSS Client is capable of learning which IP subnet is associated with each bridge port, and sends to each port only the broadcast traffic directed to the associated subnet.

It is important to mention that in our example, although logical routing between two such subnets is performed by the central router (MPS), actual forwarding of frames can still be done at the edge of the network (at the 8270s). This means that if a frame is flowing from IP subnet 8.8.1.x to 8.8.4.x, for example, and if the data rate exceeds the configured threshold, then the MPC in the MSS Client will try to establish a shortcut to prevent all this traffic from flowing into the ATM to the MSS Server and then back to the same 8270/MSS Client. But as soon as the MSS Client realises that the destination of the shortcut is actually itself, it will establish a *local shortcut*, thus actually switching frames from one token-ring port to another. Here we have an example of Layer 3 Switching taking place entirely within the ATM edge switch.

3.1.2 DPF VLANs

Dynamic protocol filtering (DPF) is a method by which a bridged network may be partitioned into several protocol-specific Virtual LANs, or PVLANs. Its purpose is

to limit the scope of frames that are normally forwarded over all active spanning tree ports.

Although Release 1.0 of the MSS Server had many filtering capabilities, the filters had to be configured manually. DPF was released in MSS 1.1 and its capacity has been expanded in successive releases. DPF dynamically turns filters on and off, on a port basis, based upon the traffic on each bridge port. The bridged network can thus be partitioned into (overlapping) protocol-specific subnetworks. DPF is an implementation of VLAN support based on protocol and subnet.

Dynamic protocol filtering monitors traffic over each bridge port, learning the protocols and subnets being used on that port. For the initial release of DPF, the supported protocols were IP, IPX and NetBIOS. The user may configure multiple IP and IPX subnets for DPF. For each configured subnet, the subset of bridge ports on which traffic for that subnet is being received is referred to as the forwarding domain of that subnet. DPF manages the forwarding domains for each subnet. Broadcast and multicast frames for a particular subnet will not be forwarded on bridge ports that are not in the forwarding domain of that subnet.

The user may configure multiple IP subnet PVLANs. All configured subnets should be local to the VLAN bridge (a router should not be required to reach the subnet). A port's membership in the forwarding domain of an IP subnet PVLAN is based on the source IP address of received IP ARP frames. The protocol filters are only applied to IP ARP frames; all other IP frames are bridged in the normal manner.

Also, multiple IPX network PVLANs can be configured. The source IPX network address of frames with a broadcast or multicast destination MAC address is used to determine the port membership in the forwarding domain of IPX network PVLANs. Multiple IPX networks are permitted on an ELAN as long as they use different encapsulations. IPX frames may also have an unspecified source network address (source network address field equals 0). When the source network is unspecified, DPF assumes that the source network is the unique IPX network on that port using the frame's encapsulation. If no IPX network on that port uses the frame's encapsulation, then the frame is discarded.

3.1.3 Configuring MPOA for IP and IPX

To obtain a network like the one depicted in Figure 19 on page 46, the following steps have to be followed:

- Configure MSS Clients and 8270s for SRB between the token-ring segments and the ATM ELAN. The configuration parameters for the base switch can be found in:
  - *8270 Switch Planning and Installation Guide,* GA27-4145 or
  - *8272 LAN Switch Module Planning and Installation Guide,* GA27-4163
- The MSS Client configuration parameter are explained in:
  - *Nways Multiprotocol Switched Services Family Clients Interface Configuration and Software user's Guide,* SC30-3966
  - *Nways Multiprotocol Switched Services Configuring Protocols and Features,* SC30-3819
- Configure an MPOA client for IP and/or IPX in the MSS Clients. The MPOA IPX configuration is described in 9.2.2, "MPOA Client for IPX Configuration" on

page 187. Configuring the MSS Client for MPOA is also described in *MSS Release 2.1, Including the MSS Client and Domain Client,* SG24-5231.

- Configure the MSS Server for routing IP/IPX. Remember that the LEC in TKR_1 has to be configured as a one-armed router for the two IP/IPX subnets, in other words it needs multiple IP/IPX addresses defined on the single interface.
- Configure the MPOA Server for IP or IPX in the MSS Server. The configuration details for IPX are shown in 9.2, “MPOA for IPX Configuration and Use” on page 179.

No special setup needs to be done in the MSS Client or the 8270 to allow for local shortcuts. It is built in the MPC code.

A description of all the actions needed in each machine follows:

3.1.3.1 8270 1 and MSS Client 1

Configure the 8270 so that ports connected to each of the token-ring segments are in different domains.

Configure the MSS Client with one interface in each domain, and one LEC in TKR_1. SRB has to be configured on these three interfaces with the proper segment numbers.

Do not assign any IP/IPX address to any of the interfaces since we are going to bridge these protocols through the MSS Client. Assign an IP address to the 8270 for management purposes only using IP Host Services.

Configure and enable DPF in the MSS Client for IPX and/or IP VLANs.

Configure and enable the MPOA client for the ATM interface to support IP and/or IPX.

3.1.3.2 8270 2 and MSS Client 2

The configuration for 8270_2 and MSS Client_2 are the same as above, except that in our example only one IP/IPX subnet is present at the token-ring segment and therefore there is no need to configure DPF VLANs.

3.1.3.3 MSS 1

The MSS will perform routing and MPOA Server functions. In our example it is also acting as the LAN Emulation Server, but this is not required. LES/BUS and LECS can reside elsewhere.

Configure one LEC in each ELAN (TKR_1 and TKR_2) with the right IP/IPX addresses on them: two addresses in TKR_1 and one in TKR_2. This enables the routing functions between these two interfaces in the MSS.

If IPX is to be used, it must also be enabled at box level, and the appropriate IPX circuits have to be defined.

Configure and enable the MPOA Server function for IP and/or IPX.

3.1.3.4 Endstations

No special configuration is required on the endstations; as usual with IP it is necessary to define the appropriate IP address and mask (255.255.255.0) and

set MSS_1 as the default gateway. For IPX, remember to enable source routing in the IPX clients and servers.

3.1.4 Operation

Two examples of data transfer will be shown. IP protocol will be used in both cases, but results are equally applicable to IPX now that MPOA for IPX is supported in the latest release of MSS code.

3.1.4.1 Data Transfer across the Two 8270s. Use of the MPOA Shortcut

Let us suppose that station PC102 in Figure 19 on page 46 wants to start an IP session to station PC105. Since the default gateway for station PC102 is the MSS Server (which has been configured with IP address 8.8.1.1), station PC102 will send an ARP frame asking for the MAC address of the router.

The MSS Client DPF function will learn through this ARP frame that subnet 8.8.1.0 is present on port 2 of the bridge[1].

DPF has already learned of the presence of the MSS Server and has therefore registered the fact that subnet 8.8.1.0 is also present on the LEC port, port 1. This came about because the MSS Server sent a *gratuitous* ARP broadcast when the LEC interface became active specifically for this purpose. Otherwise this port would remain blocked and the ARP request would not be forwarded across this interface.

The ARP request will therefore reach the MSS Server, the ARP response will flow back to PC102 through the MSS Client and normal IP communication will commence: PC102 starts sending IP packets to PC105 through the MSS Server. If the data transfer rate exceeds the configured threshold in the MPOA Client it will try to establish a shortcut to the egress MPOA Client MSS Client_2 in 8270_2.

When this shortcut is established, all IP traffic from PC102 to PC105 will flow through this shortcut, avoiding the MSS Server. Inside the ATM network this traffic is switched as ATM cells through hardware switches and never passes through the routing engine in the MSS Server, increasing throughput and preventing possible congestion.

The shortcut established will be using LANE encapsulation for transport of the frames (see 1.3.2, “LANE Encapsulation and Vendor Extensions” on page 27). This makes it possible for the egress MPOA Client (MSS Client_2) to forward the frames received using its hardware switch, creating a switched path from the ATM LEC port to the token-ring port without the intervention of software.

For the traffic in the reverse direction to flow through this SVC a different shortcut has to be established in the reverse direction.

3.1.4.2 Data Transfer between Different IP Subnets in the Same 8270. Use of Local Shortcuts

Now, let’s suppose that station PC103 wants to send IP data to station PC102. These two stations are on the same broadcast domain but on different IP subnets.

[1] For our example, MSS Client1’s port 1 is the LEC, port 2 is connected to token-ring #004 and port 3 is connected to token-ring #005.

Traffic from IP subnet 8.8.4.0 to 8.8.1.0 has to flow through a router, which in our example is located in the ATM cloud.

Station PC103 will send an ARP frame asking for the MAC address of its default gateway (8.8.4.1 in MSS Server 1).

The MSS Client_1 will learn from the ARP frames that IP subnet 8.8.4.0 is present on port 3 and port 1.

Once station PC103 has the router's MAC address, it will send all traffic to subnet 8.8.1.0 to this router.

The routed packets have to go into the bridge, to the ATM cloud, and then be routed at the MSS and go back through the same path to the 8270 again, to end up in the port connected to token-ring segment #004. Although ATM networks are very fast, this is clearly not the optimal path. The MSS Server could become a bottleneck in very heavy traffic situations, slowing down the whole ATM network.

In this scenario, MPOA can also be useful. Once again, when the MPC in MSS Client_1 detects that the data transfer from station PC103 (IP address 8.8.4.103) to station PC102 (IP address 8.8.1.102) has exceeded a configured threshold it will try to establish a shortcut to the ATM exit point nearest to the endstation in order to bypass the router hop.

When the MPOA Resolution Request is processed and the reply gets back to the MPC, it realises that the ATM target is actually its own ATM address. Then a local shortcut will be established so that data flows directly from port 3 to port 2 without ever entering the ATM network.

Since local shortcuts do not flow through any VCC, you cannot check them using the `LIST-VCC` command. To check that local shortcuts are established, use the command `LIST-ENTRIES` to view the ingress and egress cache tables. When an IP address is reached through a local shortcut, the appropriate field shows it. Figure 20 on page 52 shows two shortcuts in place in the egress MPOA Client running in MSS Client_1:

1. An entry which corresponds to a "real" MPOA shortcut set up across the ATM network from the other 8270, 8270_2
2. A local shortcut for traffic originating on another LAN port on the same 8270

```
MSS Client_1 MPC EGRESS>LIST-ENTRIES 8.8.4.103
Destination Protocol Address Mask [255.255.255.255]?

     Egress Cache Entries matching 8.8.4.103/255.255.255.255 :

  1) Address/Mask: 8.8.4.103/255.255.255.255   Entry Type: 1483 (HOST, DIRECT)
   LEC #:  1    Cache ID: xD   State: ACTIVE
   MPS: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00
   Source: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
   Remaining Age (mins:secs): 38:33
   Recvd Octets: N/A
   Recvd Frames Forwarded: N/A
   Recvd Frames Discarded: N/A
   Tag Value: N/A       Local Shortcut: FALSE
   DLL Header: x004040000103000ac00082101701 06b000510010aaaa030000000800
   LANE Extensions in last Imposition reply:  Formats 7, 11, 13, 17


  2) Address/Mask: 8.8.4.103/255.255.255.255   Entry Type: 1483 (HOST, DIRECT)
   LEC #:  1    Cache ID: x11   State: ACTIVE
   MPS: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00
   Source: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF
   Remaining Age (mins:secs): 37:23
   Recvd Octets: N/A
   Recvd Frames Forwarded: N/A
   Recvd Frames Discarded: N/A
   Tag Value: N/A       Local Shortcut: TRUE
   DLL Header: x004040000103000ac00082101701 06b000510010aaaa030000000800
   LANE Extensions in last Imposition reply:  Formats 7, 11, 13, 17
```

Figure 20. Checking Local Shortcuts

3.1.4.3 IP VLANs

Both data transfers studied above started with an ARP frame. This is a broadcast frame and would normally be forwarded through all the active spanning tree ports in the switch.

By the use of IP VLANs and DPF we have reduced the broadcast domain for this kind of frame to the forwarding domain of each IP subnet VLAN. This means that, for example, the ARP frame sent from station PC103 asking for IP address 8.8.4.1 is only forwarded on port 1, but not on port 2, since port 2 is only included in the forwarding domain of IP VLAN for subnet 8.8.1.0 (see Figure 21 on page 53).

In our particular example, since the IP subnets are clearly associated with ports, the IP VLANs could have been configured in a static way, assigning and excluding the ports to the VLANs manually at configuration time, instead of letting the DPF function do it by itself. However, if some stations were then moved from one token-ring segment to the other, the definition of forwarding would then have to be changed.

```
MSS Client_1 ASRT VLAN console>LIST IP ALL-SUBNETS

   ---------------------------- IP VLANS -----------------------------

   Subnet Address              = 8.8.1.0
   Subnet Mask                 = 255.255.255.0
   Port  1 (Interface  1)      = Auto-Detect and Include, Forwarding
   Port  2 (Interface  2)      = Auto-Detect and Include, Forwarding
   Port  3 (Interface  3)      = Auto-Detect and Include, Not Forwarding
   Age (expiration in minutes) = 10
   IP-Cut-Through Status:
     Tx From This VLAN    = Disabled  Reception By This VLAN = Disabled
     Packets Transmitted  = 0         Packets Received       = 0
     Tx Packets Discarded = 0         Rx Packets Discarded   = 0
   Tracking of Mac Addresses   = Disabled
   VLAN Status                 = Enabled
   Packets Processed           = 358
   Discards Due To Exclusion   = 0
   VLAN Name                   = IP Subnet 1
   ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
   Subnet Address              = 8.8.4.0
   Subnet Mask                 = 255.255.255.0
   Port  1 (Interface  1)      = Auto-Detect and Include, Forwarding
   Port  2 (Interface  2)      = Auto-Detect and Include, Not Forwarding
   Port  3 (Interface  3)      = Auto-Detect and Include, Forwarding
   Age (expiration in minutes) = 10
   IP-Cut-Through Status:
     Tx From This VLAN    = Disabled  Reception By This VLAN = Disabled
     Packets Transmitted  = 0         Packets Received       = 0
     Tx Packets Discarded = 0         Rx Packets Discarded   = 0
   Tracking of Mac Addresses   = Disabled
   VLAN Status                 = Enabled
   Packets Processed           = 627
   Discards Due To Exclusion   = 0
   VLAN Name                   = IP Subnet 4
MSS Client_1 ASRT VLAN console>
```

Figure 21. IP VLANs Forwarding Domains

3.2 Redundancy and Resilience

This solution builds on the previous example and shows an example of how the redundancy features within the MSS Server and Client can be used to create a network with no single point of failure.

The example uses an additional backup MSS and a second 8270 with MSS Client on one LAN segment. Restrictions on the number of 8270s available to us prevented us from installing a resilient 8270 on the other LAN segment.

Note

It is not possible to provide a resilient MPOA solution by installing two MSS Clients in a single 8270 because of the restriction that only one MSS Client in an 8270 can have bridging configured.

New redundancy features of MSS Release 2.2 are used in this solution; for detailed configuration information for the new features refer to Chapter 8, “MSS Release 2.2 LANE Redundancy Enhancements” on page 151.

3.2.1 A Resilient MPOA Network

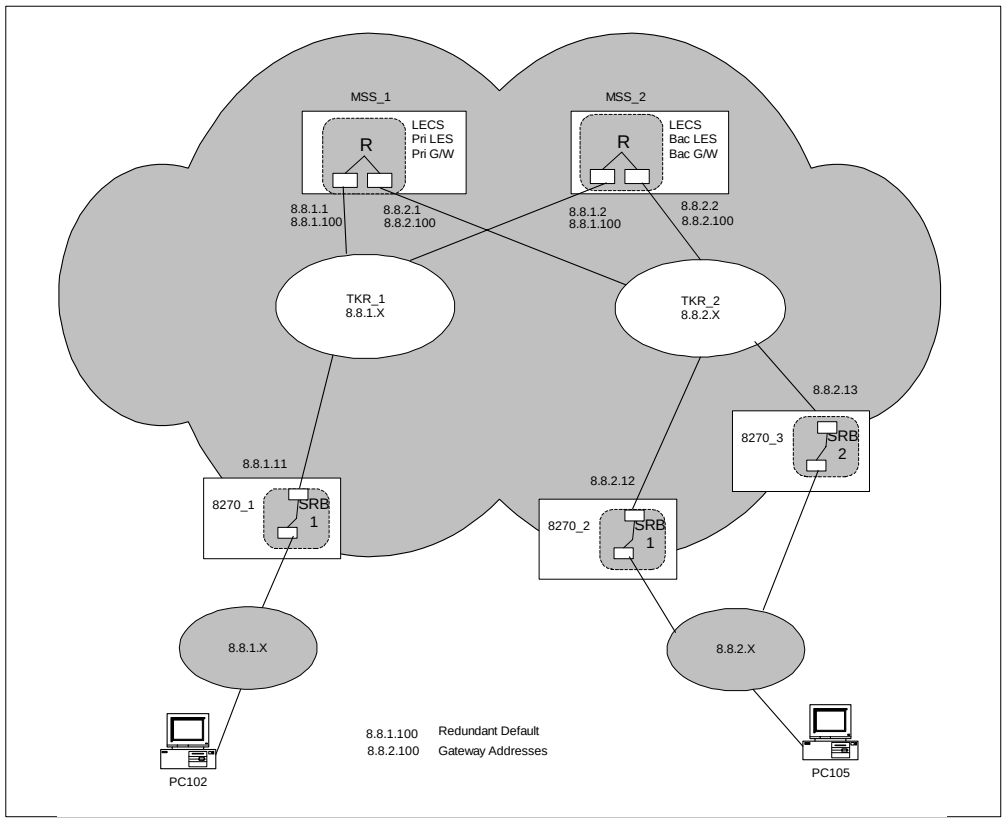


Figure 22. Solution 2: A Network with Resilience Built In

This solution provides resilience for:

The MSS:

- LECS
- LES/BUS
- Router
- MPOA Server

The 8270 (on TKR_2):

- LAN switch
- ATM uplink
- MPOA Client

3.2.2 Configuration Steps

The configuration of the resilient solution is the same as the previous example with the following additions:

MSS_1

1. Configure the two ELANs as primary and give the ATM address of the backups on MSS2 (an alternative would be to configure MSS1 as primary for TKR_1 and MSS2 as primary for TKR_2).
2. Configure a primary redundant default gateway address.
3. Configure the LE Clients in *persistent data direct mode*.

MSS_2

1. Configure a LECS.
2. Configure the two ELANs as local; redundant; backup, and give the ATM address of the primaries on MSS_1.
3. Configure a LEC for each ELAN, in persistent data direct mode.
4. Configure IP in the two LE Clients.
5. Configure a backup redundant default gateway address.
6. Configure an MPOA Server.

MSS Client (8270_3)

1. Configure as a duplicate of 8270_2, but with a different ESI and MAC address for the LEC and a different bridge number.
2. Configure LE Clients in persistent data direct mode.

MSS Client (8270_1, 8270_2)

1. Configure LE Clients in persistent data direct mode.

General

1. Configure all devices with the relevant redundant default gateway.

3.2.3 MSS Resilience

To demonstrate the MSS resilience features in this solution, three different failure scenarios were simulated:

1. Resilience features configured, except persistent LE Clients, no MPOA.
2. Resilience features configured, including persistent LE Clients, no MPOA.
3. Resilience features configured, including persistent LE Clients, with MPOA.

In each configuration example two IP pings were established continuously:

1. Ping 1 - between workstation PC105 and MSS_2 (on the same ELAN).
2. Ping 2 - between workstation PC102 and PC105 (on different ELANs and different IP subnets).

The following failure situations were then simulated and the results noted:

1. MSS_1 failure
2. MSS_1 recovery

3.2.3.1 Scenario 1 - No Persistent LE Clients, No MPOA

1. MSS_1 failure

Both pings stopped for approximately 30 seconds and then continued.

MSS_1 was running the LES/BUS function for both ELANs and the default gateway router between them. When MSS_1 failed MSS_2 took over these functions. This takeover took 30 seconds during which time the MSS Client LE Clients were disconnected from an ELAN and could not pass data.

2. MSS_1 recovery

   Both pings continued without interruption.

   Because peer redundancy was configured, the LES/BUS functions remained on MSS_2. The redundant gateway router function returned to MSS_1, but as the backup did not relinquish control until the primary was ready for operation, there was no traffic loss. In fact, there could be a 5 second gap during which neither the primary nor the backup gateway was active, but in fact the backup gateway will continue routing traffic until all Data Direct VCCs to it time out.

3.2.3.2 Scenario 2 - Persistent LE Clients, No MPOA.

1. MSS_1 failure

   Ping 1 (same ELAN) continued without interruption.

   Ping 2 (different ELANs) stopped for approximately 20 seconds and then continued.

   MSS_1 was running the LES/BUS function for both ELANs and the default gateway router between them. When MSS_1 failed MSS_2 took over these functions. Ping 1 continued because the persistent LE Clients were able to pass traffic while the LES/BUS function moved to MSS_2. Ping 2 was unable to continue despite the persistent LE Clients because the default gateway router failed; MSS_2 took over this function after 20 seconds and the ping continued.

2. MSS_1 recovery

   Both pings continued without interruption.

   Because peer redundancy was configured, the LES/BUS functions remained on MSS_2. The redundant gateway router function returned to MSS_1, but as the backup did not relinquish control until the primary was ready for operation, there was no traffic loss.

3.2.3.3 Scenario 3 - Persistent LE Clients, MPOA.

1. MSS_1 failure

   Both pings continued without interruption.

   MSS_1 was running the LES/BUS function for both ELANs and the default gateway router between them. When MSS_1 failed MSS_2 took over these functions. Ping 1 continued because the persistent LE Clients were able to pass traffic while the LES/BUS function moved to MSS_2. Ping 2 was using an MPOA shortcut between MSS Clients and was therefore unaffected by the temporary unavailability of the default gateway router and the persistent LE Clients were able to pass traffic while the LES/BUS function moved to MSS_2. The MPOA shortcuts will not remain in place, though; the MPOA server regularly sends *keep-alive* messages to the MPOA Client, and the MPOA Client will stop using shortcuts when it stops receiving these keep-alive messages. In our case, by the time the ingress MPOA Client noticed the

absence of keep-alive messages, the backup default gateway router was active and available for traffic.

In the above example if MSS_2 did *not* have a default gateway router configured, traffic would initially continue to be “routed” across the short-cut. However, the ingress MPOA client would detect the absence of a route and disconnect the shortcut.

In configurations where there were more than one MPOA server in the path between the MPOA clients, if one server failed, the other would detect this and immediately disconnect the shortcut with a *purge request.*

2. MSS_1 recovery

   Both pings continued without interruption.

   Because peer redundancy was configured, the LES/BUS functions remained on MSS_2.

3.2.4 MSS Client Resilience

To demonstrate the resilience provided by installing a second MSS Client and 8270 in this solution, a ping was established between workstation PC102 and PC105. Failure of the MSS Client which was passing the ping traffic was simulated.

This solution not only provides resilience but also increased throughput and load balancing between the MSS Clients due to the operation of source route bridging which is used in the MPOA configuration.

1. MSS Client_2 failure

   Ping stopped for five minutes and then recovered.

   When workstation PC102 sent the initial IP ARP for its gateway it cached both its MAC address and the Routing Information Field (RIF). Subsequent pings were sent only on the path described in that RIF.

   When MSS Client_2 failed, there was an alternative path available for the ping through MSS Client_3, however MSS Client_3 had a different bridge number and therefore the path did not match the path in the cached RIF. The ping was therefore unable to use the available path until the workstation timed out the cache and sent a new IP ARP, at which point the alternative path was discovered and the ping continued.

   The time taken to recover from this failure scenario is dependent on the IP implementation in the workstation. If the ARP cache is cleared manually (or the workstation rebooted!) the ping will continue immediately.

2. MSS Client_2 recovery

   Ping continued without interruption.

   For the reasons given above the ping will continue to use MSS Client_3 and will ignore the availability of MSS Client_2.

The exhibited behaviour is identical to that expected when two parallel source route bridges are installed in a “legacy” network.

3.2.5 Conclusion

When MPOA is used in conjunction with the resilience functions of the MSS and MSS Client a highly resilient network can be created, providing, in most cases, recovery from component failure that is transparent to the user. It should be noted however that limitations of the network protocols used by the client workstations will sometimes require network sessions to be re-established.

3.3 MPOA across Mixed-Media Networks

Our last example deals with mixed-media networks. It is common to find networks in which ATM backbones are used to interconnect token-ring and Ethernet LAN segments. Previous examples only dealt with token-ring to token-ring shortcuts, not least because until now the only IBM implementation of MPOA has been in token-ring edge switches.

A new IBM Ethernet switch, the 8371 Multilayer Ethernet Switch, will provide the MPOA client for IP and IPX in an Ethernet environment. This switch was generally available on April 23, 1999. For more information about the 8371 Multilayer Ethernet Switch see 2.2.5, “8371 Multilayer Ethernet Switch” on page 40.

In this example we will show how shortcuts between an MSS Client running in an 8270 token-ring switch and the MSS Client running in a 8371 Multilayer Ethernet Switch are established. Specific details about how to configure the 8371 for MPOA are given, although it should be noted that the configuration details differ little from the equivalent details for the MSS Client on the 8270 token-ring switch.

3.3.1 MPOA between Ethernet and token-ring networks

The network we are using is depicted in Figure 23 on page 59. We are using one MSS Server to route between IP subnets and to act as the MPOA Server. The 8371 switch connects the Ethernet LAN to ATM and acts as an MPOA Client. The 8270 with MSS Client installed bridges the token-ring segments to the ATM ELAN and also acts as an MPOA Client.

Shortcuts between Ethernet and token-ring are established just as would have been the case between two token-ring edge devices. There are some aspects that have to be taken into consideration:

- **Frame Size:** The maximum frame size in token-ring is larger than in Ethernet. This means that for frames traversing from the token-ring side to the Ethernet side, some fragmentation has to take place. When frames travel in the reverse direction, no fragmentation is needed.

  The 8371 does not support fragmentation. It was felt that frames originating from the Ethernet LAN would not need fragmentation, but it imposes the requirement on the MSS Client in the token-ring edge switch to perform fragmentation if necessary. The correct *fragmentation mode* has to be chosen when configuring the MPOA Client in the MSS Client.

- **Frame Encapsulation:** Token-ring and Ethernet use different encapsulation formats, so some translation has to be done. The router performs this translation and informs the MPOA Client of the DLL header that has to be used. This is done in the MPOA Imposition Request. Depending on the type of short-cut being used (see 1.3.1, ““Tagged” and “Non-Tagged” Frame Formats”

on page 26 and 1.3.2, “LANE Encapsulation and Vendor Extensions” on page 27) the job of putting the new header in the frame is done in different MPCs:

- IBM-proprietary LANE Shortcuts: It is the ingress MPOA Client which has to provide the DLL header, so that the egress MPC receives the frame as if it came from a normal VCC and has only to switch the packet between ATM and LAN. This has two advantages:

  1. Allows the establishment of shortcuts to non-MPOA LAN Emulation clients.
  2. Allows the egress MSS Client to make better use of its hardware and switch outgoing frames in hardware.

- 1483 Non-Tagged: The egress MPOA Client receives the frames in the VCC and inserts the DLL header in the outgoing frame, according to the information received in the cache imposition request.
- 1483 Tagged: The egress MPOA Client does the same work, but this time, the ingress MPC is required to put a tag into the frame. The egress MPC uses this tag to speed up location of the imposed cache entry containing the DLL header for the frame. This results in a much faster processing of the frame at the egress MPC.

  The 8371 is optimised for this last type of shortcut and will always try to receive a shortcut of this kind.

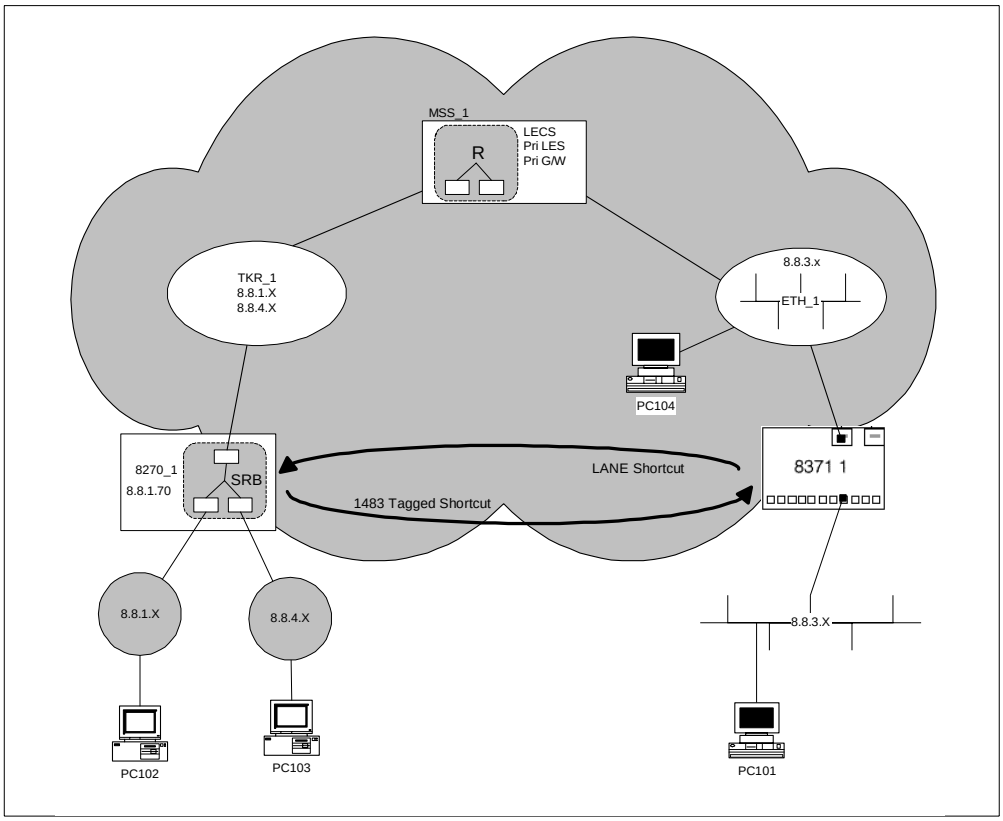


Figure 23. MPOA between Ethernet and Token-Ring Networks. Use of the 8371 Switch

- **Shortcut VCCs:** Shortcuts are always unidirectional. Put another way, for data to flow in both directions, two shortcuts are always needed. However, in theory

both shortcuts can share the same VCC. It can sometimes happen that shortcuts are established using two different VCCs[2]. If this happens, after a while, both VCCs will collapse into one single VCC for which both shortcuts flow.

In a network such as ours, the shortcuts that are set up between 8371 and 8270 take different formats:

- The 8371 acting as egress MPOA Client indicates its preference for RFC 1483-Tagged shortcuts in the Cache Imposition Reply, and will cause the 8270 acting as ingress MPOA Client to store this tag value and use this format.
- The 8270 acting as egress MPOA Client indicates its preference for LANE Encapsulation which, because the 8371 supports this IBM extension format, is used in the other direction.

Don't Panic!

The preceding discussion on the specifics of shortcut encapsulation formats is required in order to explain the details in some of the subsequent displays. It all happens quite automatically, and is simply a consequence of the different implementations of MPOA in the two different types of device. It is totally invisible to the end users, and the real message is that MPOA works between Ethernet and token-ring edge switches without any special configuration considerations.

3.3.2 Configuring MPOA across Mixed-Media Networks

To configure the example shown on Figure 23 on page 59, the following steps have to be followed:

1. Configure MSS Client and 8270 for SRB between the two token-ring segments and the ATM token-ring ELAN.
2. Enable the MPOA Client for IP in the MSS Client. Remember to set the fragmentation mode to *perform fragmentation.*
3. Configure the MSS Server for routing between the IP subnets.
4. Enable the MPOA Server for IP in the MSS Server.
5. Configure the 8371 for TB between the Ethernet ports and the Ethernet ELAN.
6. Enable MPOA Client for IP in the 8371.

Only the last two steps will be covered here. Steps 1 through 4 have already being covered in this chapter or are explained in later chapters.

Only a basic introduction to 8371 configuration will be given. For detailed information refer to *8371 Networking Multilayer Ethernet Switch Installation and Planning Guide, GA27-4226*.

3.3.2.1 8371 configuration

The 8371 builds on the same code as the MSS, called Common Code, and the following configuration steps will be familiar to anyone already accustomed to the implementation of the MSS Client in the 8270. The following pages show all the steps used to configure the 8371 and the commands used:

[2] This can happen when two MPCs are both in the process of establishing shortcuts to each other at the same time.

1. Set the ATM End System Identifier (ESI) and assign LEC to interface ATM/0.

   A LEC must be assigned to a physical interface in order for it to join an emulated LAN. In the 8371 LECs are always created at boot time. We don't need to create them, only to configure them according to our needs.

```
Config>NETWORK 36
ATM user configuration
ATM Config>ASSIGN-LEC
select LEC to assign [-1] ? 40
ATM Config>INTERFACE
ATM interface configuration
ATM Interface Config>ADD ESI
ESI in 00.00.00.00.00.00 form []? 40.00.83.71.00.01
ATM Interface Config>EXIT
ATM Config>
```

Figure 24. 8371 LEC Assignment

2. Configure LEC1 and enable all interfaces.

   The appropriate parameters have to be configured into the LEC. All interfaces are disabled by default, so we have to enable them.

```
ATM Config>LE-CLIENT
ATM LAN Emulation Clients configuration
LE Client config>CONFIG
Emulated LAN interface number [40]?
ATM LAN Emulation Client configuration
Ethernet Forum Compliant LEC Config>SET ELAN-NAME ETH_1
Ethernet Forum Compliant LEC Config>SET ESI-ADDRESS
Select ESI
   (1) Use burned in ESI
   (2) 40.00.83.71.00.01

Enter selection [1]? 2
Ethernet Forum Compliant LEC Config>SET SELECTOR
Selector byte for primary ATM address in hex [3]? 11
Ethernet Forum Compliant LEC Config>SET MAC-ADDRESS
Use adapter address for MAC? [Yes]: NO
MAC address [00.00.00.00.00.00]? 40.00.83.71.11.01
Ethernet Forum Compliant LEC Config>EXIT
LE Client config>EXIT
ATM Config>EXIT
Config>ENABLE INTERFACE 36
Interface enabled successfully
Config>ENABLE INTERFACE 40
Interface enabled successfully
```

Figure 25. 8371 LAN Emulation Client Configuration

3. Configure the MPOA Client.

Only one MPOA Client instance per ATM interface is allowed. The MPOA Client is created by default. We need to enable and configure it.

```
Config>PROTOCOL MPOA
Multi Protocol Over ATM user configuration
MPOA config>MPC

MPOA Client user configuration
==============================
ATM Device Number for the MPC [36]?
MPC/36 >CONFIG
MPC/36 Configuration>SET SELECTOR
Selector Byte (in hex) [2]? BF
MPC/36 Configuration>SET ESI
        [1] Burned in ESI
        [2] 40.00.83.71.00.01
 ESI:  [1]? 2

MPC/36 Configuration>SET SHORTCUTS
Enable LANE Shortcuts?  [Yes]?

  Choices for Source MAC Address for LANE Shortcuts:
        [1] Burned in ESI
        [2] Locally Configured MAC Address
        [3] MAC Address from the Resolution Reply

MAC Address Type for LANE Shortcuts:  [1]? 2
MAC Address for LANE Shortcuts:  [00.00.00.00.00.00]? 40.00.83.71.22.01
MPC/36 Configuration>SET IPX-PROTOCOL
Enable IPX (Yes/No):? [No]: Y

MPC/36 Configuration>ENABLE

    MPC is already ENABLED
MPC/36 Configuration>
```

Figure 26. 8371 MPOA Configuration

4. Set HOST IP Address, write config and reboot.

Since no IP address is defined on the interfaces, we will use IP Host Services to manage the box.

```
MPC/36 Configuration>EXIT
MPC/36 >EXIT
MPOA config>EXIT
Config>PROTOCOL HST
TCP/IP-Host Services user configuration
TCP/IP-Host config>SET IP-HOST
IP-Host address [0.0.0.0]? 8.8.3.71
Address mask [255.0.0.0]? 255.255.255.0
IP-Host address set.

TCP/IP-Host config>EXIT
Config>WRITE
Config Save: Using bank B and config number 2
Config>
*RELOAD
Are you sure you want to reload the gateway? (Yes or [No]): Y
```

Figure 27. 8371 Host Services User Configuration

At this point the machine is ready to work as an MPOA Client and to establish shortcuts with the MSS Client running in the 8270 token-ring switch.

3.3.3 Verifying the Operation

In this section we will verify that MPOA shortcuts are established between token-ring and Ethernet MPOA Clients. Some important aspects about these shortcuts will be highlighted.

To verify shortcut establishment, the same commands as for the MSS Client are used. First we will check the cache entries (for ingress and egress MPOA Clients) and verify that our stations are in the list. This means that we have a shortcut established and it is being used.

Then it is time to check the VCCs that were created and which shortcuts flow through them. Results for the MSS Client in the 8270 and the 8371 will be shown so that the reader can easily check how the shortcut goes from one MPOA Client to the other.

In our test network a configuration change had to be made before any MPOA shortcuts were established. The data transfer threshold default value of 10 frames per second is probably realistic for real networks, but for our test environment we wanted to reduce this value to force the creation of shortcuts. Had we not made this change, our traffic volumes would not have triggered the creation of any MPOA shortcuts, and all the traffic would have followed the "default path" through the MSS acting as a router. This change is shown in Figure 28 on page 64. Since MPOA is dynamically reconfigurable, there is no need to reboot the machine. The changes are made using *Talk 5* and they take place immediately; if the machine is reset the changes are lost.

```
+PROTOCOL MPOA
MPOA>MPC

MPOA Client Console
==================
ATM Device Number for the MPC [36]?
MPC/36 >CONFIGURE

MPOA Client Dynamic Configuration
=============================
MPC/36 CONFIGURE>FRAME-COUNT
    Frame Count for Shortcut Setup (in frames):  [10]? 1
MPC/36 CONFIGURE>EXIT
MPC/36 >
```

Figure 28. Changing the Transfer Rate Threshold

1. Checking the MPC configuration in the 8371. This shows that the MPOA Client is enabled and that the Transfer Rate Threshold has been changed by the previous step.

```
MPC/36 Configuration>LIST

   MPC Configuration
   -----------------
   STATUS: ENABLED
   Shortcut Setup Frame Count:       1     (frames)
   Shortcut Setup Frame Time:        1     (sec)
   Initial Retry Time:               5     (sec)
   Maximum Retry Time:               40    (sec)
   Hold Down Time:                   160   (sec)
   VCC Timeout Period:               20    (min)
   Accept Config From LECS:          Yes
   Fragmentation Mode:               Maximize Shortcut Usage

   Interface:                        36
   ESI:                              40.00.83.71.00.01
   Selector:                         0xBF
   Desired PCR:                      155000 (kbps)
   Maximum Reserved Bandwidth:       155000 (kbps)
   Line Rate:                        155 (Mbps)
   Enable LANE Shortcuts:            TRUE
   Source MAC Address for Shortcuts: Locally Configured
                                     40.00.83.71.22.01

   IP-Protocol: ENABLED
   IPX-Protocol: ENABLED
```

Figure 29. Show the 8371 MPOA Client Configuration

2. Checking the MPOA Server Neighbours. This shows the ATM address of the MPOA Server running in our MSS Server with its associated MAC address.

```
MPC/36 >NEIGHBOR-MPSs

MPOA Client MPS Console
=======================
MPC/36 MPS>LIST
          List of Neighbor MPSs for MPOA Client (interface 36):
          ================================================
 1) Control ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00

     1 MAC Address(es) Learnt For This MPS:

      1) MAC Addr: x40.00.82.10.19.02   Associated LEC Intf #: 40

MPC/36 MPS>EXIT
```

Figure 30. Verify MPOA Servers

3. Displaying the Ingress Cache entries in the 8371. This shows the IP addresses for which valid flows (in excess of our configured threshold) have

been detected and whether corresponding shortcuts have been set up or resolved.

```
MPC/36 >INGRESS-CACHE

MPOA Client Ingress Cache Console
=================================
MPC/36 INGRESS>LIST
          IP-Ingress Cache For MPC on ATM Interface 36
          ===========================================


          Ingress Cache Entries for Direct Host Routes:
          ---------------------------------------------
   1)  IP Address: 9.24.105.113          State: FLOW DETECTION
   2)  IP Address: 9.24.104.215          State: FLOW DETECTION
   3)  IP Address: 8.8.2.105             State: RESOLVED
   4)  IP Address: 9.24.104.108          State: HOLD DOWN
   5)  IP Address: 9.24.105.3            State: RESOLUTION

          Ingress Cache Entries for Direct Network Routes:
          ------------------------------------------------

          Ingress Cache Entries for Derived Host Routes:
          ----------------------------------------------

MPC/36 INGRESS>
```

Figure 31. Display 8371 Ingress Cache Entries

4. Displaying the Egress Cache entries in the 8371. This shows the cache entries that correspond to IP destinations on the Ethernet LAN segment.

```
MPC/36 >EGRESS-CACHE

MPOA Client Egress Cache Console
================================
MPC/36 EGRESS>LIST
          Egress Cache For MPC on ATM Interface 36
          ========================================

      Egress Cache Entries w/ MPOA-Tag Encapsulation:
      -----------------------------------------------

   1) IP Address/Mask: 8.8.3.102         /255.255.255.255      State: ACTIVE
   2) IP Address/Mask: 8.8.3.103         /255.255.255.255      State: ACTIVE



      Egress Cache Entries w/ Native 1483 Encapsulation (Host Routes):
      ----------------------------------------------------------------

   1) IP Address/Mask: 8.8.3.71          /255.255.255.255      State: ACTIVE
   2) IP Address/Mask: 8.8.3.103         /255.255.255.255      State: ACTIVE


      Egress Cache Entries w/ Native 1483 Encapsulation (Netwk Routes):
      -----------------------------------------------------------------



MPC/36 EGRESS>
```

Figure 32. Display the 8371 Egress Cache Entries

5. Viewing shortcuts in the 8371. This shows two VCCs in addition to the one used for connection to the MSS Server; the first displayed shows the LANE encapsulation format VCC set up towards the 8270 and shows that traffic destined for IP address 8.8.2.105 is using this shortcut (this corresponds to the “RESOLVED” entry for 8.8.2.105 in Figure 31 on page 66). The second displayed VCC shows the RFC 1483 encapsulation format VCC set up in the opposite direction from the same 8270.

```
MPC/36 >VCCs

MPOA Client VCC Console
=======================
MPC/36 VCC>LIST
          SVCs For MPC On ATM Interface 36 (total    2):
          ========================================
   1) VPI/VCI 0/434    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00
   2) VPI/VCI 0/435    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
   3) VPI/VCI 0/436    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF

MPC/36 VCC>LIST-VCC 0 435

  VPI/VCI: 0/435     State: OPERATIONAL    Calling Party: TRUE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:4544/4544
  Remote ATM Addr: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: TR-LANE
  H/W Path Valid: FALSE    Ref. Frame Cnt: 0
  Frames Tx/Rx: 17/0


      (Direct) Shortcut Routes Using This VCC:
      ----------------------------------------
     1) Address/Mask: 8.8.2.105/255.255.255.255    State: RESOLVED


MPC/36 VCC>LIST-VCC 0 436

  VPI/VCI: 0/436     State: OPERATIONAL    Calling Party: FALSE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:1536/1536
  Remote ATM Addr: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: LLC 1483
  H/W Path Valid: FALSE    Ref. Frame Cnt: 0
  Frames Tx/Rx: 0/53


      (Direct) Shortcut Routes Using This VCC:
      ----------------------------------------

MPC/36 VCC>
```

Figure 33. Viewing Shortcuts in the 8371

6. Viewing shortcuts in the MSS Client on the 8270. Again, this shows two separate VCCs between the 8270 and the 8371 and shows that traffic destined for IP address 8.8.3.103 is using the RFC 1483 encapsulation format for the shortcut to the 8371 Ethernet edge switch.

```
MSSC_2 MPC VCC>LIST
         SVCs For MPC On ATM Interface 0 (total    5):
         ======================================
   1) VPI/VCI 0/245    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00
   2) VPI/VCI 0/248    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00
   3) VPI/VCI 0/249    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF
   4) VPI/VCI 0/250    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.83.71.00.01.BF
   5) VPI/VCI 0/253    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.83.71.00.01.BF

MSSC_2 MPC VCC>LIST-VCC 0 250

  VPI/VCI: 0/250     State: OPERATIONAL    Calling Party: FALSE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:4544/4544
  Remote ATM Addr: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.83.71.00.01.BF
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: TR-LANE
  H/W Path Valid: TRUE    Ref. Frame Cnt: 0
  Frames Tx/Rx: 0/245     Bytes Tx/Rx: ?/?


      (Direct) Shortcut Routes Using This VCC:
      ----------------------------------------


MSSC_2 MPC VCC>LIST-VCC 0 253

  VPI/VCI: 0/253     State: OPERATIONAL    Calling Party: TRUE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:1536/1536
  Remote ATM Addr: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.83.71.00.01.BF
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: LLC 1483
  H/W Path Valid: FALSE    Ref. Frame Cnt: 0
  Frames Tx/Rx: 232/0     Bytes Tx/Rx: 16704/0


      (Direct) Shortcut Routes Using This VCC:
      ----------------------------------------
     1) Address/Mask: 8.8.3.103/255.255.255.255    State: RESOLVED
```

Figure 34. Viewing Shortcuts in the MSS Client on the 8270

Notes:

As can be seen by points 5 and 6, there are two VCCs between the MSS Client and the 8371. Each one of them carries one shortcut.

- VCC 0.435 shown in Figure 33 carries the shortcut from the 8371 to MSS Client (ultimately to station 8.8.2.105). This shortcut was established by the 8371. The encapsulation type is TR-LANE, as was explained above, to take advantage of the MSS Client capabilities of switching the packets it receives through hardware.

- VCC 0.250 shown in Figure 34 is the other end of this VCC. As we can see the frame size is 4544, which is required for the token-ring segment. Since the VCC encapsulation is LANE, and this is the egress point, the field H/W Path Valid is set to TRUE, meaning that the MSS Client is taking advantage of its ability to switch these frames in hardware.

  The MSS Client on the 8270 has no shortcut mapped into this VCC since it was the 8371that started it.

- VCC 0.253 in Figure 34 carries the shortcut from the MSS Client on the 8270 to the 8371 (to station 8.8.3.103). This VCC is using encapsulation type “Tagged 1483", as is required by the 8371 at the setup phase.

  The frame size is 1536, which means that the MSS Client on the 8270 is performing fragmentation.

- VCC 0.436 in Figure 33 shows the other end of this VCC. Now the 8371 is not aware of the shortcut, since it was established by the MSS Client. All that the 8371 has to do is insert the right DLL header into the frame and forward it to the Ethernet LAN segment. It does this by looking at the tag (provided by the MSS Client) and at information stored in its cache.

- Looking at Figure 31 and Figure 32, we see that there is an entry in the Egress cache for the destination IP address 8.8.3.103, and an entry in the Ingress cache for the source IP address 8.8.2.105, as expected.

3.3.4 Conclusion

In this example we have set up an scenario using both token-ring and Ethernet networks. By using the 8371 and the MSS Client, shortcuts are automatically established between the two different media types.

In particular the following results were proved:

1. Shortcuts between token-ring and Ethernet networks are possible, even using different frame sizes in both networks. The MSS Client is the entity in charge of fragmenting the IP token-ring frames.
2. When establishing short-cuts between one 8371 and one MSS Client (8270) the following shortcuts are established (pinging from the token-ring side to the Ethernet side):
   - The shortcut from token-ring to Ethernet is started by the MSS Client but uses the format specified by the 8371 because of the Cache Imposition Reply. Therefore it is a tagged 1483 shortcut with frame size 1536.
   - The shortcut from Ethernet to token-ring is started by the 8371 and uses the format specified by the MSS Client. Therefore it is a LANE shortcut with a frame size of 4544.

Part 2. MSS V2.2 New Features and Enhancements

Chapter 4. MSS Product Overview

This chapter describes the products which are part of the MSS family. It then reviews the various software and hardware versions that have been released and indicates the compatibility between them. The final section looks at the Configuration Tool V2.2 and explains the configuration conversion process necessary for migrating from earlier releases.

4.1 MSS Family Introduction

This section provides a brief overview of the inter-relationship between members of the MSS family. For a more detailed discussion of each product refer to Chapter 2, “Layer 3 Switching Implementations” on page 31.

4.1.1 MSS Server

The MSS Server provides the network designer with a focal point on which to configure all the major components of an ATM LANE or CIP network, including powerful bridging and routing support. Configuring these components in a single place simplifies the design and implementation of the network, and allows sophisticated redundancy features to be implemented.

4.1.2 MSS Domain Client

The MSS Domain Client resides in the LAN switch. It is a one-slot module with no ATM interface port. Its primary role is to provide the LAN switch with the capability of:

- Edge routing
- Layer 3 routing between LAN domains and ATM

4.1.3 MSS Client

The MSS Client resides in the LAN switch. It is a two-slot module with an ATM interface port. Its primary role is to provide the LAN switch with the capability of:

- NHRP

  Layer 3 switching via *edge routing*, using a router in each switch and ATM shortcuts between Layer 3 networks.

  or:

- MPOA

  Layer 3 switching via *virtual routing*, using a central route calculator and ATM shortcuts between Layer 3 networks.

For a full discussion on Layer 3 switching, edge routing, virtual routing, NHRP and MPOA see Chapter 2, “Layer 3 Switching Implementations” on page 31.

Note

The MSS Client and MSS Domain Client are together referred to as the *MSS Family Client.*

4.1.4 Related Products

There are other IBM networking products which are not part of the MSS family but nonetheless interact with the MSS Server and MSS Clients to extend the features described. For example, the newly announced 8371 Multilayer Ethernet Switch is capable of participating in an MPOA network without an installed MSS Client because it runs Common Code (see 4.2.1 below) with the MPOA Client software.

4.2 MSS Family History (Software)

The following chapters in Part 2 of this book discuss in detail the new features introduced with MSS Release 2.2 of the code. For completeness, we have listed below the existing features and the release of code in which they were introduced. Please refer to the previously mentioned redbooks for more detailed information.

4.2.1 Introduction

The MSS, MSS Client and MSS Domain Client share the same base software code. This code is known as the *Common Code* and is used in a number of IBM networking products including:

- 2210
- 2212
- 2216
- 8371

Use of the Common Code ensures compatibility between product, shared functionality and a similar look and feel to the management interface.

4.2.2 MSS Server

4.2.2.1 MSS Server Release 1.0 (October 1996)

Functions supported with this release of MSS are listed below:

- ELAN services
  - LAN Emulation Configuration Server (LECS)
  - (Redundant) LAN Emulation Server (LES)
    - Intelligent LES (ILES)
  - (Redundant) Broadcast and Unknown Server (BUS)
    - Intelligent BUS (IBUS)
    - BroadCast Manager (BCM)
  - ATM Forum-Compliant LE Client (LEC)
- Classical IP
  - (Redundant) ARP Server
  - LIS Client
- Routing
  - IP, IPX
- Bridging

    - Transparent Bridging (TB)
    - Source-Route Bridging (SRB)
- Source-Route Transparent Bridging (STB)
    - Source-Route Translational Bridging (SRTB)

4.2.2.2 MSS Server Release 1.1 (April 1997)

Functions supported with this release of MSS are listed below:

- ELAN services
    - Redundant Default IP Gateway for ELAN
    - ELAN QoS Support
    - BUS Forwarding Modes
- Classical IP
    - Redundant Default IP Gateway for CIP
    - Redundant ARP Server Improvement (Primary/Backup)
- New interfaces
    - ATM Virtual Interface (AVI)
    - FDDI (8210 only)
    - IBM-compliant LE Client
- Routing
    - NHRP Server/Client (Shortcut Routing) for IP
    - AppleTalk Routing
- Bridging
    - SuperELAN (Shortut Bridging) (TB only)
    - Dynamic Protocol Filtering (DPF)
    - Bridge Broadcast Management (BBCM)
    - RFC 1483 Bridging (PVCs)

4.2.2.3 MSS Server Release 2.0 (October 1997)

Functions supported with this release of MSS are listed below:

- Classical IP
    - Multicast Address Resolution (RFC 2022)
    - Distributed ARP Server
    - RFC 1577+ Clients
- Routing
    - Zero-Hop Routing Server; enables zero-hop (IP) routing for legacy LAN devices
    - RIP V2 for IP
    - IP/IPX Enhancements
    - APPN and Banyan VINES Routing
    - LAN Emulation

- Bridging
  - RFC 1483 Bridging (SVCs)
  - SR-TB Duplicate MAC Address Support
  - Dynamic (Selective) Linking and Loading (DLL)

4.2.2.4 MSS Server Release 2.0.1 (December 1997)

Functions supported with this release of MSS are listed below:

- Bridging
  - SuperELAN Bridging (TB/SRB)
  - Multiple SuperELANs Running Separate Bridge Instances
  - Bridge Broadcast Management (BBCM) Enhancements
  - BBCM
  - Dynamic Protocol Filtering (DPF) Enhancements
    - MAC Address VLANs
    - MAC Address VLANs
    - Sliding Window VLANs
    - Display VLAN Membership by MAC Address

4.2.2.5 MSS Server Release 2.1 (June 1998)

Functions supported with this release of MSS are listed below:

- ELAN services
  - LECS Access Controls
  - Enhanced QoS Support for ELANs
  - LANE V2 (LUNI V2)
  - BCM IPX Server Farm Detection
- Classical IP
  - IP Multicast over ATM (MARS Client/Server, MCS)
  - Peer ARP-Server Redundancy
- Routing
  - APPN and Banyan VINES routing on FDDI
  - APPN - Enterprise Extender (HPR over IP)
  - IP - MPOA Server (beta)
- Data Link Switching (DLSw)
- Bridging
  - Dynamic Protocol Filtering (DPF) Enhancements
    - Port-based VLANs
    - IP Multicast VLANs
  - FDDI TB Bridging
    - Requires new FDDI Adapter

4.2.3 MSS Client Previous Code Releases

The MSS Client was first released with MSS Release 2.1. The code is a subset of the server code and adopts the same release numbering scheme.

4.2.3.1 Release 2.1 MSS Client Functions:

- ELAN services
  - ATM Forum-Compliant LE Client (LEC)
  - ELAN QoS
- Classical IP
  - ARP Server
    - Peer ARP Server Redundancy
    - Redundant Default IP Gateway
    - Distributed ARP Server
  - LIS Client
    - RFC 1577+ Support
  - Multicast Address Resolution (RFC 2022)
    - MARS Client
  - ATM Virtual Interface (AVI)
  - Routing
    - IP
    - RIP V1/V2, OSPF V2, BGP V4
    - IPX, AppleTalk, Banyan VINES
- Bridging
  - Source-Route Bridging only
  - Bridge Broadcast Manager
    - IP
    - NetBIOS
  - Dynamic Protocol Filtering (DPF)
    - PVLAN (IP, IPX, NetBIOS)
    - Sliding Window VLANs
    - MAC-address VLANs
    - IP Multicast VLANs
    - VLAN IP cut-through
- Network Management
  - LNM Agent
  - Line Command Interface
  - HTTP server (for Browser Support)
  - GUI Configurator
  - SNMP Agent

4.2.4 MSS Domain Client Previous Code Releases

The MSS Client was first released with MSS Release 2.1. The code is a subset of the server code and adopts the same release numbering scheme.

4.2.4.1 Release 2.1 MSS Domain Client Functions

MSS Domain Client has the following functions:

- Routing
  - IP
    - RIP V1/V2, OSPF V2, BGP V4
  - IPX, AppleTalk, Banyan VINES
- Bridging
  - Source-route Bridging
  - Token-Ring (8270/8272) only
  - Bridge Broadcast Manager (BBCM)
    - IP
    - NetBIOS
  - Dynamic Protocol Filtering (DPF)
    - PVLAN (IP, IPX, NetBIOS)
    - Sliding Window VLANs
    - MAC-based VLANs
    - IP Multicast VLANs
  - VLAN IP cut-through
- Network Management
  - LNM Agent
  - Line Command Interface
  - HTTP server (for Browser Support)
  - GUI Configurator
  - SNMP Agent

4.3 MSS Family History (Hardware)

This section describes the history of the hardware platforms of the MSS Family.

4.3.1 Introduction

There have been a number of hardware releases of the MSS. In general these have been announced in line with the major software versions. All software releases have been backward compatible (will run on old hardware versions), however the new hardware is often required to realize some of the new functions of the software. A full compatibility chart is included later in this chapter, 4.4, “MSS Family Hardware/Software Compatibility” on page 83.

4.3.2 MSS Server Current Models

The MSS Server is currently available in three formats:

1. A stand-alone version (MSS Server 3.0, 8210-003)
2. A module to fit the IBM 8265 (MSS Server 3.0 Module, FC5403)
3. A module to fit the IBM 8260 (MSS Server 2.0 Module, FC5400)

The stand-alone version and the 8265 module are new with MSS Release 2.2 and require this software release. The module for the 8260 was released with MSS Release 2.0. All existing versions of the MSS Server can run the latest level of code provided that they have sufficient memory installed (see Table 4 on page 83).

4.3.2.1 MSS Server 3.0 and MSS Server 3.0 Module Features

The MSS Server 3.0 has the following features:

- 233 MHz PowerPC 740
  - Upgraded from the previous version's 166 MHz PowerPC 603ev
  - Includes a Superscaler giving up to four instructions per clock cycle
- 64 MB EDO DRAM
- 32 KB 8-way cache
- Charm 2.1 ATM adaptor
  - Improved interrupt management
  - PCI bus interface improvements
  - Additional adapter memory

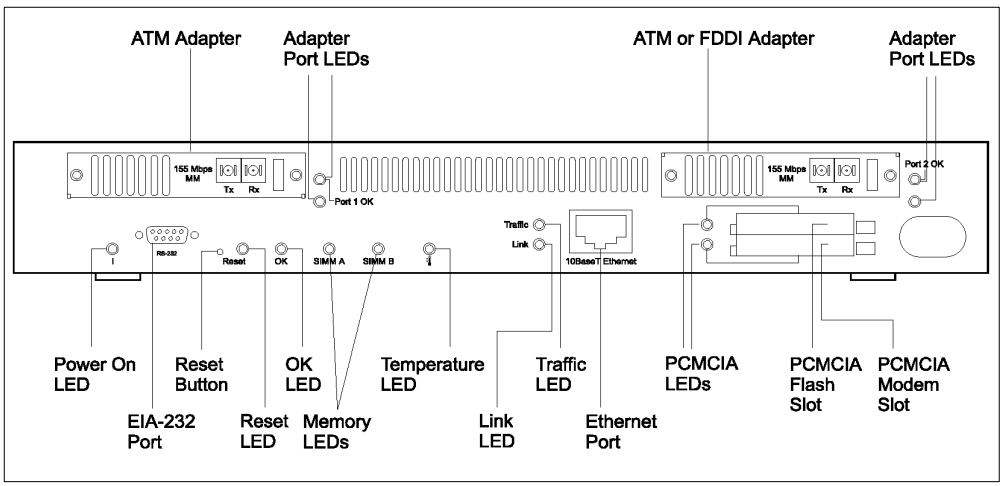


Figure 35. Front Panel of the IBM MSS Server 3.0 (8210-003)

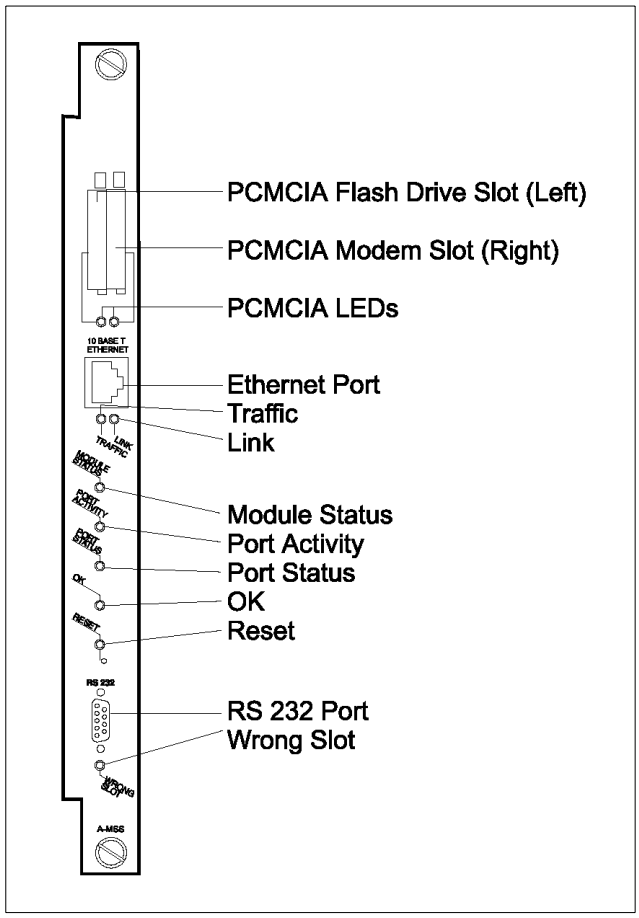


Figure 36. Front Panel of the IBM MSS 3.0 Server Module

4.3.3 MSS Server Previous Models

For comparison the following section lists the main hardware features of previous MSS versions.

4.3.3.1 MSS Server 8210-001, MSS Server Module (2 slot, 8260)

Released with software version MSS Release 1.0:

- 100 MHz PowerPC 603e
- 32 MB DRAM (field upgradeable to 64 MB)
- 512 KB L2 cache
- 12 MB flash
- PCMCIA removable hard drive
- PCMCIA modem slot

4.3.3.2 MSS Server 8210-002, MSS Server Module (1 Slot, 8260, 8265)

MSS Server Module released with software version MSS Release 2.0, MSS Server 2 released with software version MSS release 2.1:

- 166 MHz PowerPC 603eV
- 64 MB EDO DRAM

- 512 KB L2 cache
- 12 MB flash
- 1.6 GB IDE internal hard drive
- PCMCIA modem slot
- 10 Mbps Ethernet port

4.3.4 MSS Domain Client

There is only one hardware version of the MSS Domain Client. The MSS Domain Client is a single-slot Universal Feature Card (UFC) for the following platforms:

- IBM Nways 8270 Model 600/800 token-ring switch
- IBM Nways 8272 Model 216 token-ring switch
- Three-slot 8272 LAN switch module for the IBM 8260/8265
- Two-slot 8272 LAN switch module for the IBM 8260/8265

The MSS Domain Client has no ATM interface and must be used together with an ATM UFC to provide connectivity between token-ring and ATM. It has the following hardware features:

- 133 MHz PowerPC 603e
- 32 MB processor DRAM
- 8 MB flash memory (internal)
- 1 MB buffer memory SRAM for LAN ports
- Layer 2 hardware switching ASICs
  - Universal Feature Interface Chip (UFIC) for LAN switch interface and inbound frame filtering/forwarding
  - Memory Interface to Memory Interface Chip (MIMIC) for communication between the MSS Domain Client processor and UFIC.

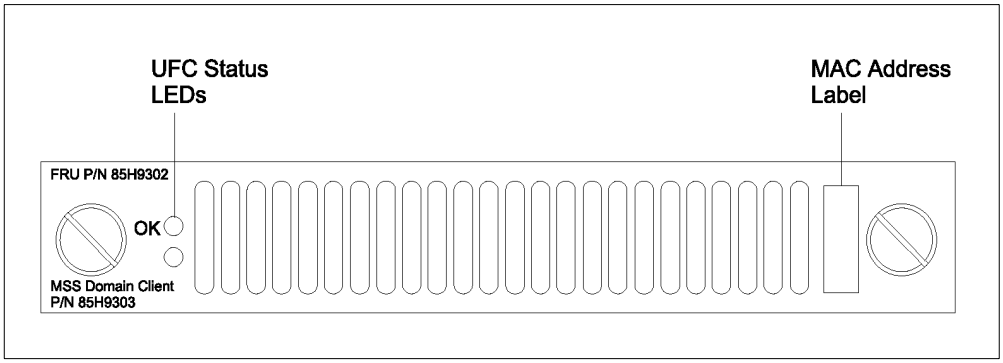


Figure 37. MSS Domain Client

4.3.5 MSS Client

There is only one hardware version of the MSS Client. The MSS Client is a two-slot Universal Feature Card (UFC) for the following platforms:

- IBM Nways 8270 Model 600/800 token-ring switch
- Three-slot 8272 LAN switch module for the IBM 8260/8265

The MSS Client has one ATM interface and can co-exist with an ATM UFC, MSS Domain Client or a second MSS Client. It has the following hardware features:

- 133 MHz PowerPC 603e
- 32 MB processor DRAM
- 8 MB flash memory (internal)
- 1 MB buffer memory (SRAM) for LAN ports
- 8 MB buffer memory per ATM interface
- Layer 2 hardware switching ASICs
  - Universal Feature Interface Chip (UFIC) for LAN switch interface and inbound frame filtering/forwarding
  - Memory Interface to Memory Interface Chip (MIMIC) for communication between the MSS Domain Client processor and UFIC.
- 155 Mbps ATM interface (MMF or SMF)

Note

The combined number of MSS Client UFCs, Domain Client UFCs and ATM UFCs in a single switch must not be greater than two. Only one may be used for bridging (and therefore for MPOA).

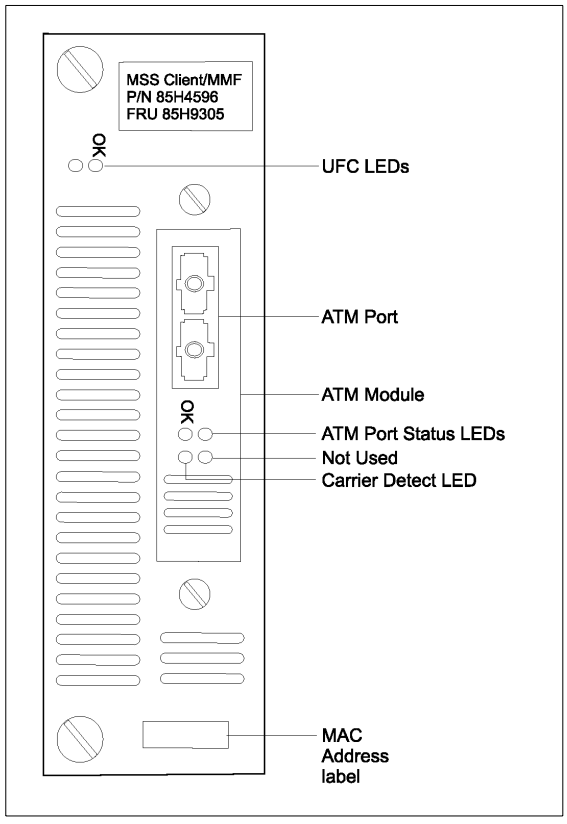


Figure 38. MSS Client

4.4 MSS Family Hardware/Software Compatibility

This section identifies the compatibility between various hardware and software releases of the MSS family products.

There are two components to the software which runs on the MSS family products:

- Firmware

  The interface between the hardware and the operational code. A new version of firmware is usually required to support new releases of hardware.

- Operational Code

  Based on the Common Code, the operational code provides all the software features of the product. New versions are released to provide additional product function and the ability to make use of new hardware.

4.4.1 MSS Server

The chart in Table 4 shows the minimum firmware version that is required for each hardware platform running each operational code level.

It may be found that unsupported combinations will run, but this may lead to unpredictable operation and is not recommended.

Note

64MB of memory is required for MSS release 2.0 or later.

Table 4. Hardware; Firmware; Operational Code Compatibility Chart

MSS Model	Description	Mem (MB)	R1.0	R1.1	R2.0	R2.0.1	R2.1	R2.2
8210-001	Stand-alone, 2U high	32	1.0	2.0	**XXXX**	**XXXX**	**XXXX**	**XXXX**
		64	**XXXX**	3.0	3.0	3.0	3.0	4.0
FC5300	8260 two-slot module	32	1.0	2.0	**XXXX**	**XXXX**	**XXXX**	**XXXX**
		64	**XXXX**	3.0	3.0	3.0	3.0	4.0
FC5400	8260/5 one-slot module	64	**XXXX**	**XXXX**	**XXXX**	3.1	3.1	4.0
8210-002	Stand-alone, 1U high	64	**XXXX**	**XXXX**	**XXXX**	**XXXX**	3.21	4.0
8210-003	Stand-alone, 1U high	64	**XXXX**	**XXXX**	**XXXX**	**XXXX**	**XXXX**	4.0
FC5403	8265 one-slot module	64	**XXXX**	**XXXX**	**XXXX**	**XXXX**	**XXXX**	4.0

The first row lists the operational code release levels; the numbers in the grid are the minimum required firmware level for each hardware version.

XXXX=not supported.

Notes

To support the MSS modules the 8260 CPSW must be V2.5.2 or higher.

To support the MSS module FC5400 the 8265 CPSW must be V3.3.4 or higher.

To support the MSS 3.0 module (FC5403) the 8265 CPSW must be V4.1 or higher.

4.4.2 MSS Family Client

There is currently only one hardware version of each of the MSS Client and MSS Domain Client. There is no chart available for compatibility between operational code versions and firmware versions. It is recommended that the firmware included in the relevant operational code zip file be used.

4.5 MSS Family Software Upgrades

Detailed information on how to upgrade the firmware and operational code of the MSS Family devices is provided in the manuals shipped on CD-ROM with the hardware and are also available on the World Wide Web. This section lists the basic steps and directs the reader to the correct manual for more information.

The latest MSS Server information, release notes, operational code, firmware, configuration tool and fixes can be found on the World Wide Web at the following URL:

```
http://www.networking.ibm.com/netsupt
```

From the above URL, you can register for e-mail bulletins containing notification of code updates and the latest MSS Server information.

To access the MSS Server Release 2.2 operational code on the World Wide Web, a valid user ID and password are required. A user ID and password are included with the MSS Server R2.2 CD-ROM.

4.5.1 Upgrading the MSS Server

The following procedure can be used to upgrade the MSS if the current firmware level is at least V3.0 and the current operational code level is at least 1.1 PTF 4.

Note

If the current firmware level is lower than V3.0 or the current operational code level is below 1.1 PTF4 see the latest release notes for special considerations.

4.5.1.1 Upgrading MSS Server Firmware

Note

MSS Release 2.2 requires firmware version 4.0 on all versions of MSS.

Firmware V4.0 is included in the MSS Release 2.2 zip file (file name *firm.ld*).

1. Load the V4.0 firmware file onto the MSS hard disk (or flash) either by upgrading the operational code to MSS Release 2.2 (see 4.5.1.2, "Upgrading MSS Server Operational Code" on page 85) or by transferring the firmware image using the Firmware menu option **Copy Remote Files**.
2. Select **Updating System Firmware** from the System Management Utilities menu.
3. Select the **Local Image File** option from the Updating System Firmware menu.
4. Enter `-a` filename`.img`, where *filename* is the name of the firmware image on the hard drive. The *-a* causes the firmware recovery block to be updated along with the normal firmware image, and is needed to make dump work correctly with MSS Server V2.2.
5. Reload the MSS.

4.5.1.2 Upgrading MSS Server Operational Code

To upgrade the MSS Server operational code:

1. Get the MSS Release 2.2 Server operational code from the CD-ROM or the above-mentioned URL.
1. Update the operational code.

   There are two ways to update the operational code:

   1. It can be done in-band using the operational code as described in the section "Performing Change Management Operations*"* in the *Nways Multiprotocol Switched Services Server Interface Configuration and Software User's Guide, SC30-3818*.
   2. It can also be done from firmware using Change Management as described in the section "Using MSS Server Firmware" in the *Nways Multiprotocol Switched Services Server Interface Configuration and Software User's Guide, SC30-3818*.

4.5.2 Upgrading the MSS Family Client

There are several methods for updating the MSS Family Client with newer versions of code:

1. Using the download UFC option from the LAN switch console.
2. Updating the image while the MSS Client is operational.
3. Using the MSS Client firmware options.

4.5.2.1 Download the UFC Option from the LAN Switch Console

Please see "Updating the MSS Family Client Images" in the *MSS Family Client Release 2.2* Release Notes for detailed instructions.

The advantages of this method are:

- Both operational image and firmware can be updated.
- Both trivial file transfer protocol (TFTP) and XMODEM are supported.
- Relatively fast. Download of the operational image should take approximately three minutes through TFTP, with another minute to load the new image.

The disadvantage of this method is:

- The MSS Family Client is reset and will not be able to participate in the network operations until the update is complete.

4.5.2.2 Updating the Image While the MSS Family Client Is Operational

Please see the *Nways Multiprotocol Switches Services Family Clients Interface Configuration and Software User's Guide, SC30-3966* for details on how to use this method.

The advantages of this method are:

- The act of downloading the image is not destructive to the operation of the MSS Family Client in the network.
- Using the TIMEDLOAD option, the MSS Family Client can be scheduled to be reloaded at a later time.

The disadvantages of this method are:

- TFTP is the only supported method of loading the image.
- MSS Family Client firmware cannot be updated using this method.
- Relatively slow. This method can take up to 16 minutes, during which the MSS Family Client is still operating. It is possible that during this time, the TFTP might encounter network problems that prevent the successful update.

4.5.2.3 Updating the Image Using the MSS Family Client Firmware

Please see the *Nways Multiprotocol Switched Services Family Clients Interface Configuration and Software User's Guide, SC30-3966* for details on how to use this method.

The advantages of this method are:

- Both operational image and firmware can be updated.
- Allows for update of the firmware recovery block.

  Note: This should only be done if instructed to do so by IBM Service.
- Relatively fast. Download of the operational image should take approximately two minutes through TFTP, with another minute to load the new image.

The disadvantages of this are:

- TFTP is the only supported method of loading the image.
- Interface accesses domain 0 (default) only.
- The MSS Family Client is reset and is unable to participate in the network operations until the update is complete.

4.6 MSS Family Configuration Tool

This section provides an overview of the MSS Configuration Tool, describes the changes to the file format in the latest release and lists the steps necessary to upgrade an existing configuration to MSS Release 2.2.

4.6.1 Introduction

In common with other IBM products based on the Common Code, the MSS Family products have a number of methods of configuration :

- Command-line interface
- Web browser
- Configuration tool

The configuration tool is the most user-friendly of the configuration methods. It consists of a stand-alone software package for AIX, Windows or OS/2 workstations. It provides a graphical user interface for creating MSS Family configurations offline, which can then be downloaded to the MSS Server or Client.

There is one configuration tool for the MSS Server and one for the MSS Family Client.

Each new major release of operational code is accompanied by a new release of the configuration tool program, providing new configuration options for new features in the code.

4.6.2 Configuration Tool V2.2 Changes

With MSS Release 2.2 the corresponding configuration tool has changed format. There are changes to the look of the tool and also to the file format.

4.6.2.1 Navigation and Configuration Window Changes

The Navigation Window and Configuration Windows now have a different layout. The new layout is logical and easy to follow. If more information is required this can be found in the *readme* and *release notes* which are part of the Configuration Tool package.

Figure 39 on page 88 shows the new top level unexploded Navigation Window for the MSS Server.

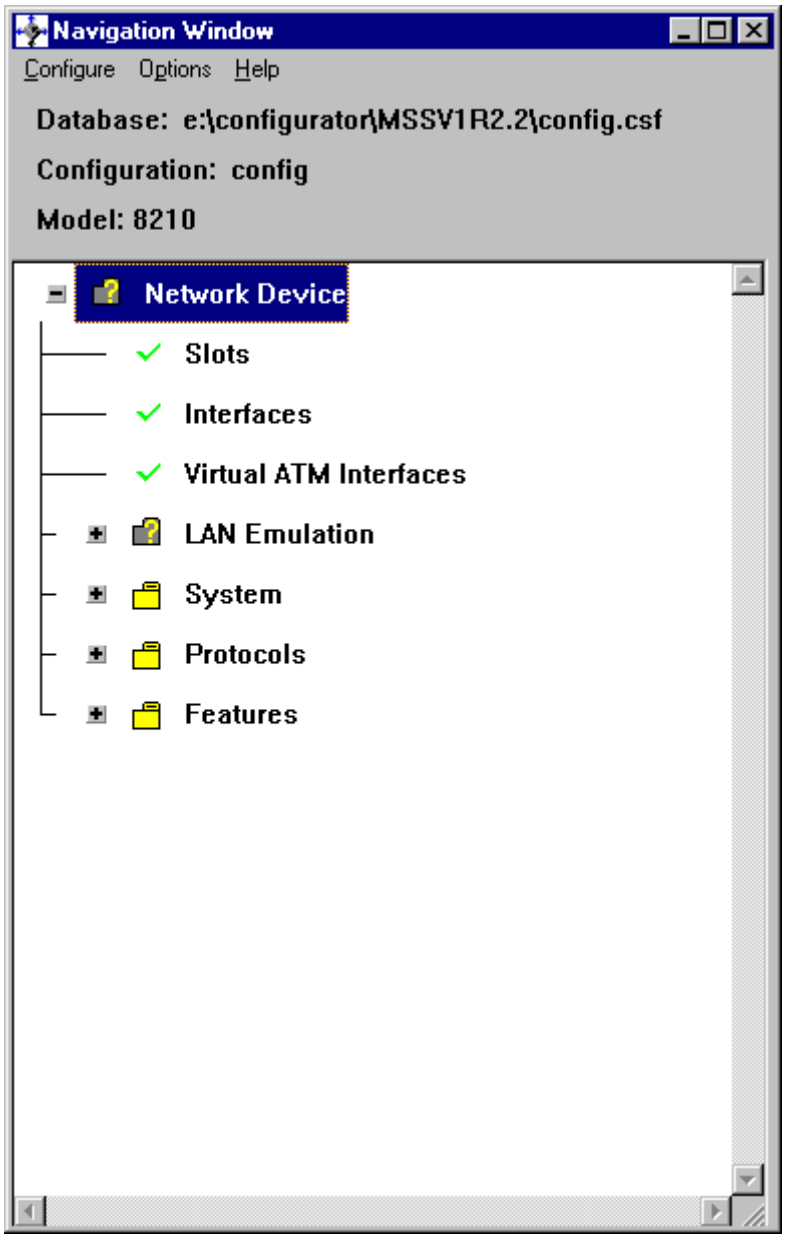


Figure 39. New Navigation Window

4.6.2.2 File Format Changes

In previous versions the configuration tool had the concept of CDB files and CFG files. The CDB files could contain a number of separate configurations in a format readable by the Configuration Tool. These configurations could be converted into CFG files executable by the MSS.

The new configuration tool no longer has the concept of CDB files; instead it has the following 2 types of files:

- CSF files

  These are files in the configuration tool format. There is only one configuration per CSF file. As with the old CDB configuration files a CSF file can be sent or

received from the MSS Server or Client using the configuration tool *communication*s option.

- CFG files

  These are in a format for execution by the MSS Server or Client. As with previous versions of the configuration tool, TFTP or XMODEM can be used to send these files to the MSS Server or Client.

4.6.3 Upgrading MSS Configurations to Release 2.2

When upgrading to a new release of MSS code it is not necessary to re-write the configuration; existing configurations can be converted to run on a new release. As when creating the original configuration there are a number of ways to achieve this. To upgrade your existing configurations using the configuration tool perform the following steps:

Note

When upgrading existing MSS family configurations to MSS Release 2.2, there is an additional step to complete because of the different file formats used by the new configuration tool.

1. If you do not have a copy of the live CDB file, retrieve one from the MSS using the *old* configuration tool (the one used to create the configuration).
2. Open the *conversion tool*, provided with the MSS Configuration Tool v2.2.
3. Select the directory (folder) containing the CDB file and the directory you wish the new CSF file to be created in.
4. Select the required configuration from the CDB file and click **Create**.

   This step converts the file format.
5. From the configuration tool open the newly created CSF file.

   This step converts the configuration to a Release 2.2 configuration.
6. Send the file to the MSS using the *communications* option or create a CFG file using the *create router configuration* option.

Note

The preceding process is only required to convert an existing configuration prepared using an earlier release of the configuration tool to one compatible with the latest version of the configuration tool. There is no need to go through a conversion process for the operational configuration when the MSS code is upgraded to the latest level; the operational configuration used for the previous level of code can be used by the latest level of code directly.

Chapter 5. MSS V2.2 New Features and Enhancements Summary

This chapter is a summary of the new features and enhancements that are available in this new release of MSS, applicable to MSS Server, MSS Client or MSS Domain Client.

These enhancements are divided into six major categories:

1. Performance Improvements
2. Network Performance Enhancements
3. LANE Redundancy Enhancements
4. MPOA Enhancements
5. Usability Enhancements
6. Miscellaneous Common Code Enhancements

A short description of each topic is given here. The following chapters will deal in more detail with each of the items mentioned here.

5.1 MSS Performance Improvements

The following enhancements are aimed to improve the performance of the MSS by itself. They do not change the way MSS behaves in the network but rather how it processes the packets internally.

Some of the performance improvements are due to new hardware, such as the support for the new Charm 2.1 PCI adapter, while others boost the performance of the MSS by rewriting the code and optimising it where possible.

If the new software Release 2.2 is used with the old MSS hardware then this optimised code will still increase MSS performance, but the functions that apply to the new hardware will be deactivated.

Even without new hardware, one of the key design aims of the new release of software has been to restore performance to at least the levels seen on Release 1.1of MSS software.

ATM Net Handler Performance Improvements - This restores lost performance. MSS performance has significantly degraded over the last several releases. Changes have been made in ATM buffer usage and caching to return the performance its former level.

Fast Path for Source Routed and 802. 3 Frames - The current LEC (LAN Emulation Client) code requires the LEC to look up low level header information for these frame types every time it sends a frame. Now the LEC can build and cache layer 2 headers, using the pre-built headers for subsequent packets to the same destinations. This results in a performance enhancement for these frame types.

Charm 2.1 Support - Charm 2.1 runs substantially faster than previous versions. The Charm device driver has been modified to obtain maximum performance with the new hardware.

Interface Receive Buffers - With the new Charm 2.1 hardware it is now possible to increase the number of receive buffers up to 1000, as opposed to 250 which was the previous maximum value. Although this change can also be applied to earlier releases of hardware, it is probably inappropriate to increase the number of buffers this much for anything other than the latest hardware, which is potentially fast enough to fill all available buffers. The new hardware also makes better use of these buffers and allows for a faster way to move packets into MSS memory.

5.2 MSS Network Performance Enhancements

The following enhancements modify the way that the MSS behaves in the network. They are aimed to reduce broadcasts in the network and therefore to increase the performance of the network as a whole.

BUS Filter - Enables filtering of packets at the BUS (Broadcast and Unknown Server) in a LAN Emulation environment. Users can filter packets based on MAC address, protocol, or define their own filters using a sliding window. This gives increased security and performance of the network.

BUS Police - Administrators can set the utilisation threshold of the BUS (Broadcast and Unknown Server) in a LAN Emulation environment. If a user of the BUS sends a number of frames per second that exceeds a threshold then the frames from that destination will be filtered. The option exists to exclude certain MAC addresses from this policing. The policing gives the network some protection from broadcast storms and denial of service attacks. The BUS Monitor function has been modified so that time between sampling intervals is now defined in seconds, rather than in minutes. This change is more appropriate for the needs of the BUS Police function.

IP Multicast VLANs - Reduces unnecessary forwarding of IP multicast data frames on the ASRT bridge or on a SuperELAN bridge. IP multicast VLANs are created automatically once this feature is enabled. This feature was announced for MSS Release 2.1 but not delivered and has finally been included in this release.

5.3 Usability Enhancements

Under this heading we group all the enhancements that try to ease the use, configuration and manageability of the MSS.

Command Completion - Lists all possible extensions to a command entered on the consoles of the MSS Server and MSS Client. This feature makes the interface of the MSS products similar to that of the 826x products. The OPCON console menu has also been reordered and the `ping` command has been introduced for convenience.

Secondary ELS Console - A new process has been created to allow access directly to a secondary event logging system (ELS) console.

Packet Trace Decoding Aids - Will decode Ethernet and token-ring Layer 2 information as well as identify the higher level protocol. It is also possible to decode some information from common frames (for example, ARP).

Dynamic 1483 PVC/ SVC - Allows activation of configured Classical IP permanent PVC/SVC entries without having to reboot the box or to reset the interface. Prior to this design, if a permanent PVC/SVC was configured in Talk 6, then the box needed to be rebooted or the interface reset to activate the configured entry(s).

Non-Zero VPI - The previous implementation of the Charm device driver only allowed PVCs to be created with a virtual path identifier (VPI) of zero. Some ATM service providers assign non-zero PVC VPIs to clients connecting to their system. Now users can specify a VPI in the range of 0-7 when configuring a PVC.

CPU Performance Monitor - Allows monitoring of the CPU load. Can be used as a tool for trend analysis, bottleneck evaluation, and capacity planning. This code was formerly made available as beta code in Release 2.0.1 PTF 4 but was not formally announced at that time, nor was it documented anywhere.

The following Usability Enhancements are taken from the common code base:

Additional Dynamic Reconfiguration - Adds the ability to reset an interface or a protocol without rebooting the MSS. It increases the support provided in previous releases.

Logging and ELS Enhancements - Restructures the ELS console to increase buffering and reduce the number of messages that are flushed and not displayed.

IP Ping Data Option - Lets users specify a pattern to include in the data portion of a ping packet.

APPN Serviceability - Adds items for tracing and debugging APPN problems.

5.4 LANE Redundancy Enhancements

This section covers all the enhancements that deal with Redundancy and LAN Emulation Services.

LES/BUS Enhanced Redundancy - Improves availability in Emulated LANs using Redundant LES/BUS by reducing the maximum amount of time required to detect the failure of a partner LES/BUS.

LES/BUS Peer Redundancy - Minimizes disruptions in a LAN Emulation network when a LES/BUS (LAN Emulation Server/Broadcast and Unknown Server) fails. Previously, a single failure would result in two outages: one when the primary failed and another when the primary restored service and the backup yielded to the primary. Now the backup and the primary are able to negotiate roles based on the number of clients each is serving at the time when the primary restores service. The goal is to disrupt the least number of clients.

LECS Database Synchronization - Eases user configuration of backup LECS (LAN Emulation Configuration Servers). The user can configure one server on one MSS and then use the ATM network to replicate the configuration on other LECS.

Persistent Data Direct VCCs - Normally when a LAN Emulation Client (LEC) loses its connection to the LAN Emulation Server (LES), the LEC immediately terminates all VCCs to other LECs. Now the user has the option of allowing the

VCCs between LECs to continue to pass traffic, even while the LEC is not connected to a LES. This will allow traffic to flow when the LEC momentarily loses contact with the LES or while the LEC is switching from a primary LES to a backup LES.

5.5 MPOA Enhancements

MSS Release 2.2 supports MPOA for IPX. Following are the modifications included in the code. Note that NHRP for IPX is not supported even though MPOA actually relies on NHRP to communicate between MPOA Servers.

MPOA Server for IPX - Adds support for IPX to the existing MPOA Server. This will allow the server to supply ATM address resolution information for IPX protocol addresses to clients with MPOA IPX shortcut capability.

MPOA IPX Client - Adds support for IPX to the existing MPOA Client. Prior to this release, only IP protocol frames could set up MPOA shortcuts. The client will attempt to create shortcuts for IPX frames of the following types: token-ring SNAP, 802.2, Ethernet SNAP, DIX, and 802.3.

MPOA Server MIB - Implements the MIB for the MPOA Server. This is done in accordance with the ATM Forum's Multiprotocol over ATM Version 1.0 MIB definition.

MPOA Client MIB - Implements the MIB for the MPOA Client. This is done in accordance with the ATM Forum's Multiprotocol over ATM Version 1.0 MIB definition.

5.6 Miscellaneous Common Code Enhancements

In this section we will cover all the common code enhancements not elsewhere covered. We cover here only the MSS-specific aspects of these.

Bridge/Route Same Protocol - Removes the restriction that does not allow routing of a protocol if it is configured for bridging on any interface. This expands the support done for MSS1.1.

IP Routing/Bridging Same Interface - Allows a bridge port to determine whether an IP frame should be routed or bridged based on the destination MAC address and eliminates the need for two interfaces on a single segment to both route and bridge IP.

IP MTU by Interface - Allows the maximum transmission unit size (MTU size) of IP packets to be configured for each interface.

IP Filter Enhancements - Provides the ability to route or drop packets based on IP addresses, ports, or interfaces.

Increase Number of Interfaces - Raises the number of interfaces supported by the router code from 254 to 16k.

DLSw Currency - Reduces the size of SRAM for DLSw and adds support for duplicate MAC addresses on different Token-rings.

OSPF Currency (RFC 2178) - Adds support for RFC 2178 which includes enhancements to OSPF authentication, addition of point-to-multipoint interface, support for overlapping area ranges and others.

Modify DVMRP Config Menus - Modifies the structures used to create DVMRP menus to make conversion to HTML possible.

Packet Tracing for Interfaces Other Than ATM - Adds support for packet tracing on non-ATM interfaces. For MSS the only type of interface affected is FDDI.

TOS - Sets the type of service field to make policy-based routing possible.

IPXWAN for Multiple DLCI - Adds no new function to the MSS but does cause the IPX menus to be restructured.

Clear IPv4 MAC Header Cache on RIF Update - Clears the IPv4 MAC header cache when a routing information field (RIF) update is detected by the interface.

APPN Configuration TG Number - Allows the configuration of APPN TG numbers for all DLC types.

APPN APING - Exposes the currently hidden APING command and redirects the output to Talk 5.

Chapter 6. Performance Enhancements

This chapter deals with the new features and enhancements which specifically affect the performance of the MSS and of the network in general.

6.1 Summary

The enhancements covered in this section are:

- Charm 2.1 Support and New Device Driver
- Software Performance Improvements:
  - ATM Net Handler Improvement
  - Fast Path for Source-routed and 802.3 IP Frames
  - Interface Receive Buffers Enhancement
- BUS Filter
- BUS Police
- IP Multicast VLANs

For a complete list of MSS 2.2 enhancements, please, see Chapter 5, “MSS V2.2 New Features and Enhancements Summary” on page 91.

The goal of the performance enhancements was to restore the performance of the MSS Server to its Release 1.1 level. Although measurements are not yet complete, it appears that the result will be a performance improvement of about 30% beyond that of Release 1.1 and that performance with small packets has been doubled.

However, the MSS Release 2.2 performance enhancements do not just increase the performance of the MSS Server itself, but also the performance of the network through the addition of LANE broadcast filters and the use of IP Multicast VLANs.

These enhancements add to the existing methods for containing broadcasts in the network such as *Broadcast Manager* (BCM), *Bridge Broadcast Manager* (BBCM) and *Dynamic Protocol Filtering* (DPF), giving the users of MSS a complete set of functions to choose from, each one best suited to a particular network topology or configuration. For a more complete description of the features available in earlier releases, please refer to:

- *MSS Release 2.1, Including the MSS Client and Domain Client,* SG24-5231
- *Understanding and Using MSS Release 1.1 and 2.0,* SG24-2115
- *Understanding and Using the IBM MSS Server,* SG24-4915

The following items are a selection of the performance enhancements in Release 2.2.

6.2 Charm 2.1 Support and New Device Driver

Scheduled to be available at the same time as the MSS Release 2.2 software is the new IBM 8210 MSS Server Model 003. This new hardware will perform a

minimum of 50% faster than older versions, due to a new RISC-based processor and a new ATM adapter: the Charm 2.1 ATM card.

This new adapter runs substantially faster and carries much more memory than previous charm adapters (16 Mb of control memory and 32 Mb of packet memory, compared to 10 Mb of total memory in the older versions). It also provides an onboard 401 Core primarily for the purpose of supporting the ATM Forum-specified Available Bit Rate (ABR) standard.

The new RISC-based processor is a PowerPC 740 processor running at 233Mhz, compared to the 166Mhz speed of previous versions.

The MSS Release 2.2 software includes support for this new ATM adapter and takes advantage of this new hardware in several ways. In addition to a bigger memory, the Charm 2.1 adapter makes better use of its memory resources and allows for direct memory access (DMA) of the received data into MSS's memory without interrupting the CPU. This removes two I/O transactions per received packet and results in better performance, especially under bursty conditions.

Also the new Charm 2.1 includes a number of fixes within the hardware driver which previously were included in the adapter device driver.

This means that to be able to use the new Model 003 hardware, MSS Release 2.2 software is needed. The converse is not true: MSS Release 2.2 can be used with older versions of hardware and all the enhancements that are not specifically related to the new hardware will still be available.

6.3 Software Performance Improvements

Apart from the hardware improvements of the new MSS Model 003, a number of modifications have been made to the Release 2.2 code to restore lost performance.

6.3.1 ATM Net Handler Improvement

Changes have been made in the ATM buffer usage and caching to return the performance to its initial level. The MSS software performance enhancements basically entail improving the L1 cache utilization.

6.3.2 Fast Path for Source Routed and 802.3 IP Frames

Another change in the MSS code includes the redesign of the way the MSS processes source-routed and 802.3 IP frames.

The current LEC transmit code path for source-routed packets (non-SR bridged packets) and IEEE 802.3 IP packets requires CPU intensive cache searches for every frame. The number of cache searches by a LEC can be greatly reduced by building, maintaining, and using a low-level header. For each destination MAC address the LEC is transmitting to, a pre-built low-level header can be used to quickly format a packet's encapsulation header. (It is the protocol code's responsibility to request, pass on transmit, and free low-level header cache entries built by the network interface).

Because the frame header does not have to be reconstructed for every transmit frame, the LEC's transmit data path is shortened and its overall performance

improved. There is no configurable or predetermined limit to the number of pre-built low-level headers that can be in use.

6.3.3 Interface Receive Buffers Enhancement

The number of Receive Buffers defaults to 80 but could be increased to a maximum of 250. This maximum number has been increased to 1,000. It's unlikely to be necessary to increase this number higher than 120 for anything other than the latest level of hardware. Even with Charm 2.1, which certainly has the capability of filling a large number of buffers quickly, increasing this value to the higher end of the allowed range may actually *decrease* performance because of an increasing number of cache misses. Any changes to the defaults should be made carefully.

6.4 Bus Data Frame Filtering

BUS data frame filtering allows specific types of data frames to be forwarded or discarded by the LANE BUS. Packets received by the BUS on a Multicast Send VCC are verified against the list of enabled filters. Packets eligible for transmission are sent on either the Multicast Send or the Multicast Forward VCC. Other packets are discarded.

This function provides customers with greater control over the type of traffic which is permitted on an ATM ELAN, improving bandwidth utilization by limiting broadcast data traffic and centralizing filter controls in a single location.

One specific example of the use of this function is as a method of filtering out traffic for specific protocols which use broadcast mechanisms for discovery. For example, it is possible to use BUS filtering to prevent NetBIOS traffic from flowing by filtering out all the explorer frames which are required before devices can communicate with each other. By preventing devices from discovering each other, no actual communication is possible even though the directed communication frames themselves do not need to pass through the BUS function.

6.4.1 Introduction

Broadcast traffic has always been a concern in all kinds of networks. Broadcast traffic has the potential for interrupting all users in the network and of affecting the overall performance.

Since ATM is not a shared medium in the sense that ATM-connected devices are not directly capable of receiving broadcast packets from other users of the ATM network, ATM networks rely on the BUS to provide the distribution services for all broadcast, multicast and unknown unicast frames to the devices in the ELAN. Because all broadcast, multicast and unknown unicast frames must flow through the BUS, the BUS is an ideal location for controlling and managing ATM backbone data traffic.

Previously, ATM LANE data filtering had to be supported and configured in the ATM edge devices. These devices provide connectivity between ELANs and Ethernet and token-ring legacy LANs. Because of the large number of ATM edge devices available, managing data traffic flows at the edge of the ATM network can be difficult and cumbersome to implement.

By using BUS Data Frame Filters, the configuration and management of backbone ATM network filters is centralised in a single MSS Server. This gives two additional benefits:

1. Network utilisation is improved by preventing unwanted data traffic from entering the ATM backbone in the first place, for example by discarding NetBIOS-type broadcasts. This results in an increase in the performance of the global ATM network. Previously, by filtering these frames in ATM edge devices, many of these frames were unnecessarily transported across the ATM network.
2. Network security is improved by restricting the data traffic allowed on the ATM backbone and the devices which participate in the data traffic.

BUS filtering expands the capabilities of the Broadcast Manager (BCM). The goal of the BCM is to learn relationships between Layer 3 addresses and MAC addresses, and therefore to convert multicast frames into unicast frames. The BUS filter simply discards unwanted traffic in the network.

6.4.2 Implementing BUS Data Frame Filters

BUS data frame filtering operates within an ELAN and is supported for both Ethernet and token-ring ELANs. All packets received by the BUS on a Multicast Send VCC are verified against the list of enabled filters. LE control frames are not filtered; neither are frames sent on a Data Direct VCC, since they don't flow through the BUS at all.

If the frame passes the BUS filtering logic, then the frame is *eligible* for transmission onto the ELAN; otherwise the frame is discarded. Note that even though a frame may pass the filtering logic it may still be discarded later for other reasons (for example by the Broadcast Manager function).

There are four types of BUS filters. Each type provides a mechanism for electing which packets pass and which are discarded. The supported filters are:

- Protocol Type (or EtherType in Ethernet networks)
- MAC Address
- IP Address
- Sliding-Window

Each configured filter may be placed on an *exclude* or *include* filter list. If the frame matches a filter on the exclude list, the frame is discarded. If the frame matches a filter on the include list, the frame passes the filtering logic and is eligible for transmission onto the ELAN. If the frame does not match any filter, then the *default* action is taken. The default action is user-configurable and can be either exclude or include.

The user can define as many filters per LES/BUS as desired. The only limitation is the amount of static (SRAM) configuration memory needed to store the BUS filter definition. It is, however, recommended not to define too many filters due to the processing overhead that each represents. There is always a trade-off between defining a few filters and to reducing broadcasts to increase the network performance and between defining too many filters which will result in the CPU being overloaded and therefore will lead to global degradation of the MSS performance.

To use BUS filtering the BUS must be configured in Adapter or System mode. VCC-Splice mode is not supported, since in this mode packets are forwarded by the ATM adapter itself and are not processed on the MSS Server's system processor.

The MSS Server also keeps track of the number of frames discarded as a result of frame filtering logic. These Filter Counters can be displayed on the console via a Telnet or Web session as part of the overall BUS statistics.

6.4.3 Configuring BUS Data Frame Filters

BUS filters can only be configured on an existing LES/BUS. You cannot configure BUS data frame filters on the MSS Client or Domain Client.

When defining filters, each configured filter can be placed on an exclude or include list. Each list can contain as many filters as are needed: the only limitation is the amount of SRAM needed to store the configuration definitions. A mixture of filter types can be configured on each list.

After defining the filters, each filter can be enabled independently for each LES/BUS and assigned to the appropriate list, with some restrictions. Defined filters can be assigned to either of the two lists in each LES/BUS, so it is possible to have the same filter in an include list in one LES/BUS and in an exclude list in a different LES/BUS. No filter can be defined in an exclude and include list at the same time in the same LES/BUS.

Note

The configuration program will prevent you from enabling the same filter in the include and exclude list of a given LES/BUS at the same time. If, on the other hand, you configure BUS data frame filtering with either the Web interface or with the command line, no such checking is performed and therefore the MSS will not prevent you from apparently enabling a filter in both lists. In such a case, only the filter in the preferred list actually takes effect.

BUS filters can be added, modified, and deleted from the MSS Server console (*Talk 5*) while the ELAN is operational. BUS data frame filtering can be enabled for each LES/BUS, and global parameters such as Preferred List Action and Default Action can be configured without the need to restart the ELAN. Any changes to the BUS filters made from *Talk 5* are lost when the MSS Server is rebooted or the ELAN is restarted.

If BUS filters are defined using the command line or with the Web interface then they have to be defined separately for each LES/BUS. Filters defined in one LES/BUS are not shared by other LES/BUS instances. If the configuration program is used, only a single filter list is needed from which each of the filters defined can later be assigned to a particular include or exclude list in a given LES/BUS. See 6.4.4, "Examples of Configuring BUS Filters" on page 106 for an explanation of the commands used to create and enable BUS filters.

If BUS filters are defined using the configuration program, then a name is given to each filter. This name is later used to refer to the filter when assigning filters to each individual LES/BUS.

6.4.3.1 Filter Lists Preference and Default Action

When both exclude and include filter lists are configured it is necessary to choose which filter list will be processed first. This can be selected for each LES/BUS individually. The corresponding configuration variable is called *Preferred filter list action*. Filters on the preferred filter list are evaluated first. Each filter in the list is evaluated sequentially. If the data frame received matches none of the filters in the preferred list, then the filters in the other list are evaluated. By default the preferred list is the exclude filter list.

If no filter matches are found after evaluating both lists, then the *default filter action* is taken. The default action is also individually configured for each LES/BUS for which the BUS data filtering option is enabled. The possible choices are include and exclude.

6.4.3.2 Supported Filter Types

There are four types of BUS filters provided. Each type provides a mechanism for electing which frames should pass or which should be discarded. Each filter list can contain one or more of any combination of filter types.

Protocol Filters

Protocol filters allow data packets to be managed by protocol type. Protocol types supported are:

- IPX
- AppleTalk
- Banyan Vines
- NetBIOS
- NetBIOS over IP
- IP Multicast
- IPX Type 20
- Bridge Protocol Data Unit (BPDU)

Other protocols can be filtered using the Sliding-Window filter.

IP Address Filters

IP filters allow any IP packet to be managed based on IP addresses or subnets. For each IP address filter defined, there is a possibility of choosing either the source or the destination IP address, but not both. To do that, two separate filters are needed. Subnet filters can be defined by setting the appropriate IP filter mask.

MAC Filters

BUS data frame filters can also be applied to the Source or Destination MAC address in the Layer 2 frame header. Each MAC address filter definition must specify if it applies to the source address or to the destination address. Both types are not allowed in the same filter; separate filters are needed.

Groups of MAC addresses can be filtered by modifying the MAC address filter mask. By default, the MAC address mask is 0x'FFFFFFFFFFFF'. If a data frame contains a RIF header, the most significant bit in the source MAC address is set to one. Source MAC Address filters applied to a token-ring BUS should always mask out this first bit using the address mask 0x'7FFFFFFFFFFF'. This prevents

having to configure two Source MAC filters, one for frames originated with a RIF header and one for frames originated without a RIF header.

The MAC address of the received packet (either source MAC address, or destination MAC address, depending on whatever has been configured) is ANDed with this mask value. The configured MAC address for the filter is also ANDed with the mask. Both results are then compared and checked to see if a match is found.

Sliding-Window Filters

Sliding-Window filters are the most powerful and flexible filters that can be defined. They allow the filtering of a packet based on any kind of data that it might contain, not being restricted to protocol type, or MAC or IP address.

Sliding-Window filters are used to scrutinise data packets based on data located at a configurable frame offset. User-definable data and masks can be used to compare data fields up to 32 bytes in length. To create a sliding-window filter, the following parameters have to be defined:

- **Packet Offset** value specifies where the requested data should be found within the frame. Sets where the sliding-window should be applied.
- **Base Offset** indicates whether the packet offset should be counted from the beginning of the MAC header or from the beginning of the I-frame.

  The beginning of the I-frame starts with the EtherType field for Ethernet DIX frames, and with the length field for Ethernet 802.3 frames.

  For token-ring frames it starts after the RIF field or after the Source MAC Address field if no RIF is present in the frame.
- **Window Data** is the actual data that should be looked for.
- **Window Mask** specifies the compare mask for the sliding-window filter. The sliding window mask is ANDed with the packet data starting at the specified offset in the packet.

All BUS filter types are stored in separate SRAM records for easier configuration management but at run time all BUS filters defined are transformed into Sliding-Window filters. This means that actually the MSS only needs to process one type of filters (sliding-window filters). The other three filter types that we have just seen are simply three kinds of pre-defined sliding-window filters that are defined for easier configuration and management.

Protocol filters, as well as IP and MAC address filters, can easily be changed into sliding-window filters, and then modified accordingly to user needs. The following tables lists the relationship between sliding-window filters and the other types of filters. Please note that some filters require the application of more that one sliding-window to identify the right frame. Also, filters are applied differently if they affect Token-ring or Ethernet ELANs, and also for Ethernet DIX and 802.3 encapsulation. The tables are included for reference purposes and also can be used as a guide to creating other sliding-window filters.

Table 5. Sliding-Window Filters (1)

Protocol	ELAN Type	Offset in I-Frame / Length	Data	Mask
IP	Eth & TR	2 & 0 /8	AAAA030000000800	FFFFFF000000FFFF
	Eth & TR	2 & 0 /8	AAAA030000000806	FFFFFF000000FFFF
	Eth (DIX)	0 /2	0800	FFFF
	Eth (DIX)	0 /2	0806	FFFF
IPX	Eth & TR	2 & 0 /8	AAAA030000008137	FFFFFF000000FFFF
	Eth & TR	2 & 0 /2	E0E0	FFFF
	Eth (DIX)	0 /2	8137	FFFF
	Eth (DIX)	0 /2	FFFF	FFFF
AppleTalk	Eth & TR	2 & 0 /8	AAAA03000000809B	FFFFFF000000FFFF
	Eth & TR	2 & 0 /8	AAAA0300000080F3	FFFFFF000000FFFF
Banyan Vines	Eth & TR	2 & 0 /8	AAAA0300000080C4	FFFFFF000000FFFF
	Eth & TR	2 & 0 /8	AAAA030000000BAD	FFFFFF000000FFFF
	Eth & TR	2 & 0 /8	AAAA030000000BAF	FFFFFF000000FFFF
IP Multicast	Eth & TR	2 & 0 /25	AAAA030000000800 4500000000000000 0000000000000000 E0	FFFFFF000000FFFF FF00000000000000 0000000000000000 F0
	Eth (DIX)	0 /19	0800450000000000 0000000000000000 0000E0	FFFFFF0000000000 0000000000000000 0000F0
IPX Type 20	Eth & TR	2 & 0 /14	AAAA030000008137 000000000014	FFFFFFFF000000FF 0000000000FF
	Eth & TR	2 & 0 /8	E0E0000000000014	FFFF00000000000FF
	Eth (DIX)	0 /8	8137000000000014	FFFF00000000000FF
	Eth (DIX)	0 /8	FFFF000000000014	FFFF00000000000FF
BPDU	Eth & TR	2 & 0 /3	424203	FFFFFF
NetBIOS	Eth (DIX)	0 /8	80D5000000F0F003	FFFF000000FFFFFF
	Eth & TR	2 & 0 /3	F0F003	FFFFFF

Protocol	ELAN Type	Offset in I-Frame / Length	Data	Mask
NetBIOS over IP	Eth & TR	2 & 0 /32	AAAA030000000800 4500000000000000 0000000000000000 0000000000890089	FFFFFF000000FFFF FF00000000000000 0000000000000000 00000000FFFFFFFF
	Eth & TR	2 & 0 /32	AAAA030000000800 4500000000000000 0000000000000000 00000000008A008A	FFFFFF000000FFFF FF00000000000000 0000000000000000 00000000FFFFFFFF
	Eth & TR	2 & 0 /32	AAAA030000000800 4500000000000000 0000000000000000 000000000000008B	FFFFFF000000FFFF FF00000000000000 0000000000000000 000000000000FFFF
	Eth & TR	2 & 0 /30	AAAA030000000800 4500000000000000 0000000000000000 00000000008B	FFFFFF000000FFFF FF00000000000000 0000000000000000 00000000FFFF
	Eth (DIX)	0 /26	0800450000000000 0000000000000000 0000000000000089 0089	FFFFFF0000000000 0000000000000000 000000000000FFFF FFFF
	Eth (DIX)	0 /26	0800450000000000 0000000000000000 000000000000008A 008A	FFFFFF0000000000 0000000000000000 000000000000FFFF FFFF
	Eth (DIX)	0 /26	0800450000000000 0000000000000000 0000000000000000 008B	FFFFFF0000000000 0000000000000000 0000000000000000 FFFF
	Eth (DIX)	0 /24	0800450000000000 0000000000000000 000000000000008B	FFFFFF0000000000 0000000000000000 000000000000FFFF

Table 6. Sliding-Window Filters (2)

IP Address	IP Frame	Encapsulation Type	Offset in I-Frame / Length	Data	Mask
Source	IP	Eth & TR (SNAP)	22 & 20 /4	User specified IP Address	User spec. mask
		Eth (DIX)	14 /4	User specified IP Address	User spec. mask
	IP ARP	Eth & TR (SNAP)	24 & 22 /4	User specified IP Address	User spec. mask
		Eth (DIX)	16 /4	User specified IP Address	User spec. mask
Destination	IP	Eth & TR (SNAP)	26 & 24 /4	User specified IP Address	User spec. mask
		Eth (DIX)	18 /4	User specified IP Address	User spec. mask
	IP ARP	Eth & TR (SNAP)	28 & 26 /4	User specified IP Address	User spec. mask
		Eth (DIX)	20 /4	User specified IP Address	User spec. mask

Table 7. Sliding-Window Filters (3)

MAC Address	ELAN Type	Offset in MAC-Hdr / Length	Data	Mask
Destination	Eth & TR	0 /6	User specified MAC	User specified mask
Source	Eth & TR	6 /6	User specified MAC	User specified mask

6.4.4 Examples of Configuring BUS Filters

In this section we will learn how to configure BUS filters. Two examples will be given.

6.4.4.1 Setting BUS Filtering to Contain Broadcast and Eliminate Unwanted Traffic

In this first scenario we have the network depicted in Figure 40 on page 108. There is one MSS Server running LAN Emulation Services and two MSS Clients are being used to interconnect LAN token-ring segments to the ATM backbone, through source-route bridging.

This network is an example of a typical ATM network with several token-ring sites connected through an ATM backbone. Remote sites are represented by the token-ring segments behind the 8270s.

The endstations are Windows 95 and Windows NT clients using IPX, IP, and NetBIOS protocols. There are also some NetBIOS and IPX Servers.

Without any filtering, all traffic will be bridged to all stations, potentially resulting in many interruptions to the endstations because of heavy broadcast traffic associated with NetBIOS and IPX.

In our example we want to keep all NetBIOS and IPX traffic within the remote sites so that the overhead of broadcast traffic does not affect the performance of the ATM backbone. We also want to have IP bridged across the ATM backbone so that all the stations can communicate with each other using IP, in this case even between different remote sites.

To make things a little more complicated, some stations directly attached to ATM need to have access to the NetBIOS servers that are located in different remote sites. These ATM stations are identified by their MAC addresses. If the stations were located on a token-ring segment instead of being directly attached to ATM the result would be the same and would make no difference in the way the filters are set up. In our example, we will have one station, PC101, that needs to access the servers. We will enable all frames for this MAC address.

These traffic requirements are shown in Figure 40 on page 108 with different arrows between the stations and servers. It is important to remember that this is just an example of what a network could look like. As many combinations of filters as are wanted can be made and you are not restricted to the configurations that we're showing.

To address this network problem, MAC filters could be used in the 8270s. This solution is actually being used in many networks but has several disadvantages compared to the BUS data frame filtering solution:

- With BUS Filtering only one filter definition is needed at the LES/BUS. This makes management and configuration much easier.
- Some devices used to interconnect ATM to legacy networks cannot support MAC filtering. With BUS Filtering, all the work is done at the BUS so any ATM bridge can be used.
- ATM stations themselves cannot be filtered.

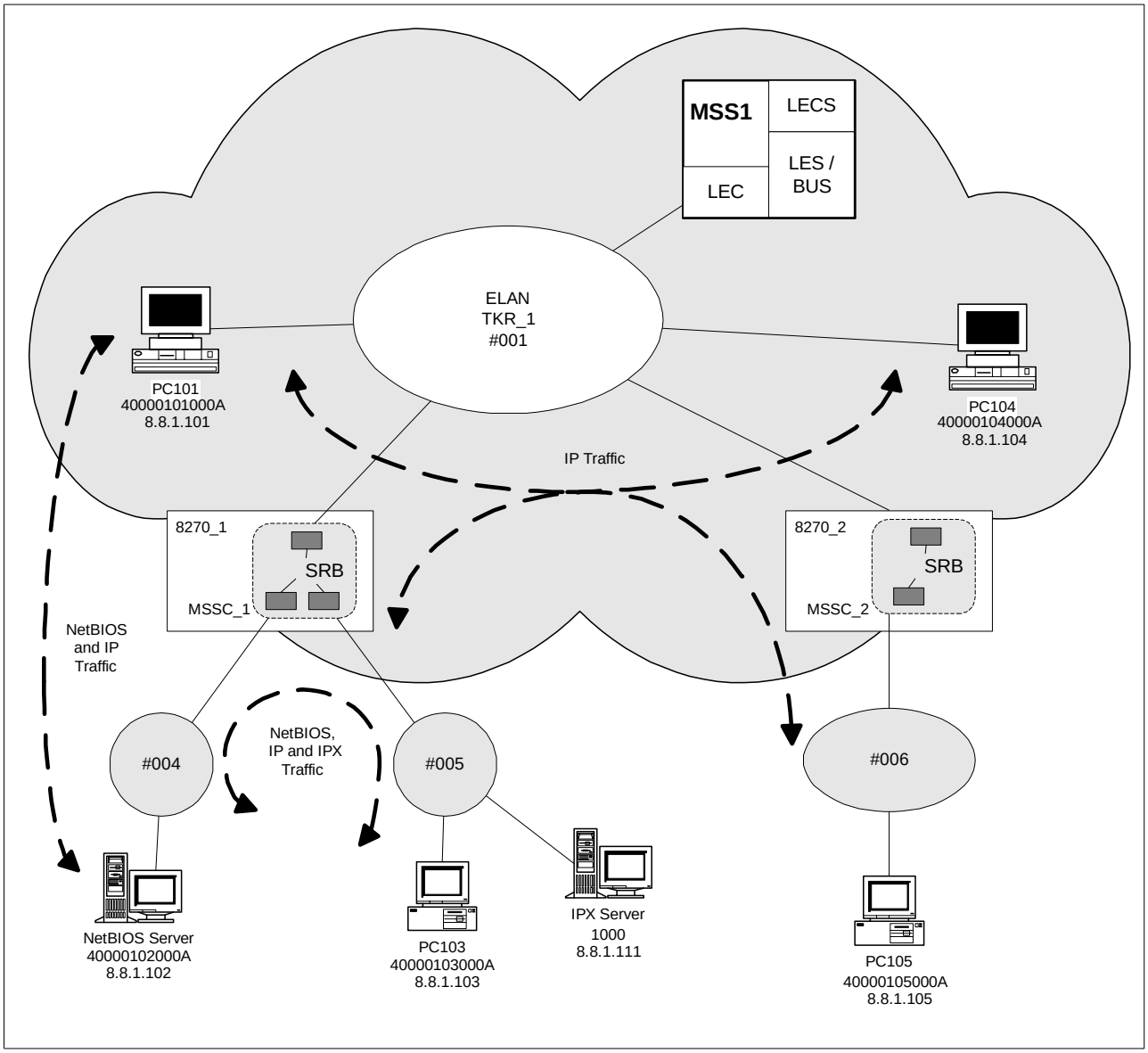


Figure 40. Network Diagram for BUS Filters Example 1

On the other hand, filtering at the edge could have some advantages over BUS filtering:

- Gives the possibility to filter traffic on a per port basis.
- Filtering functions are distributed, potentially giving higher performance and scalability capacities.

In our example, we will define some filters with the following restrictions:

1. Exclude access for NetBIOS over IP
2. Include access for IP protocol
3. Include access for certain MAC addresses to allow NetBIOS traffic
4. Include access for BPDUs in a TB environment
5. Exclude everything else

Notes:

- Point 1 is needed because the Include filter for IP will include *all* the IP protocols, including NetBIOS over IP.
- Point 3 will actually enable all protocols and frame types for the given MAC addresses, and if it were really important that only NetBIOS be allowed, then a sliding-window filter should be used which combines types of filters (MAC address and Protocol Type) into one.
- Point 4 is necessary since the last filter 5 will exclude anything not explicitly defined earlier. This means that BPDUs will be filtered out and this could create loops in a transparent bridge environment. Also, if any other type of frame is to be used, it should also be included at this stage.

6.4.4.2 Defining Protocol Filters

The first step in our example is to create the protocol filters that will be used. These filters, as explained before, should select the frames based on the protocol type, differentiating between IP frames, NetBIOS over IP and BPDU frames.

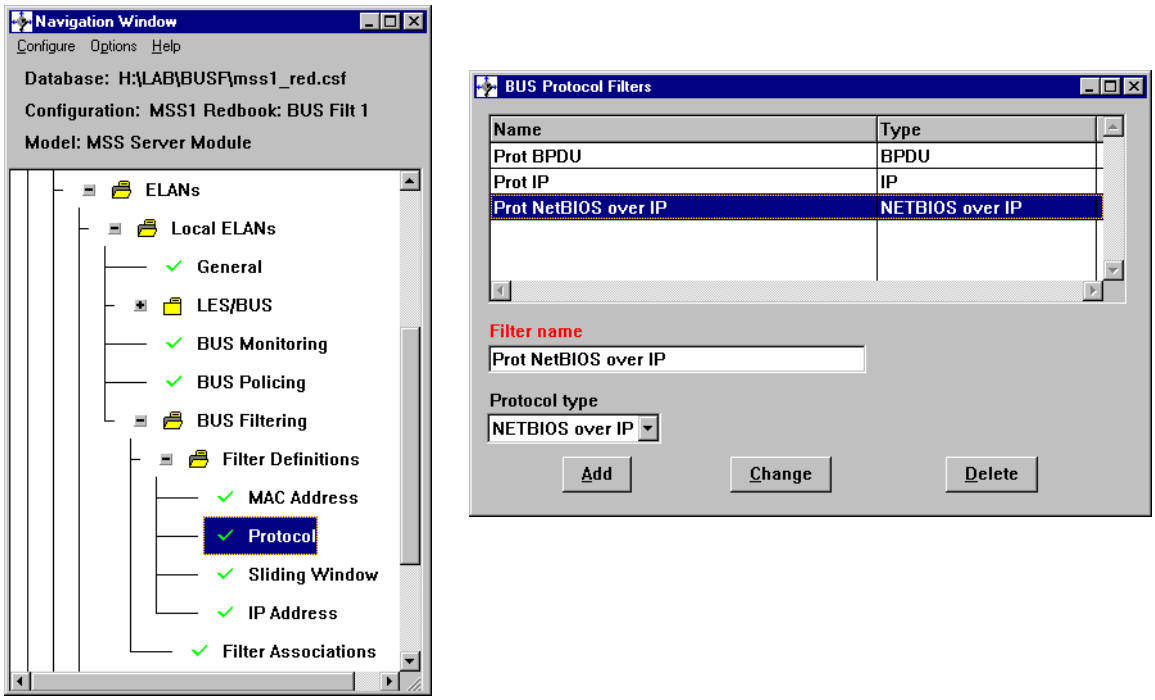


Figure 41. Defining BUS Protocol Filters

To set the filters, either *Talk 5* or *Talk 6* can be used, as can the config tool. We will use the config tool, although some of the commands that can be used in *Talk 6* will also be presented. Remember that filters defined and enabled through *Talk 5* are temporary and will be lost after the next MSS reload.

Figure 41 on page 109 shows the Protocol filters that have been defined. Defining each one is easy: simply select the appropriate protocol type from the pull-down menu, choose a name for the filter and then click the **Add** button. The equivalent commands that would have been used in *Talk 6* are shown in Figure 42 on page 110 for defining a BPDU filter.

```
MSS1 *TALK 6

MSS1 Config>NETWORK 0
ATM user configuration
MSS1 ATM Config>LE-SERVICES
LAN Emulation Services user configuration
MSS1 LE Services config>LES-BUS
  (  1) <<< New LES/BUS >>>
  (  2) ETH_1
  (  3) ETH_PROD
  (  4) TKR_1
  (  5) TKR_2
  (  6) TKR_PROD
Choice of LES/BUS [1]? 4
LES-BUS configuration
MSS1 LES-BUS config for ELAN 'TKR_1'>BUS-FILTER
BUS FILTER configuration
MSS1 BUS FILTER config for ELAN 'TKR_1'>ADD PROTOCOL
Select Protocol
        (1) IP
        (2) IPX
        (3) NETBIOS
        (4) APPLETALK
        (5) BANYAN VINES
        (6) NETBIOS over IP
        (7) IP MULTICAST
        (8) IPX TYPE 20
        (9) BPDU

Enter Selection:  [1]? 9
Select Filter List
        (1) EXCLUDE
        (2) INCLUDE

Enter Selection:  [1]? 2
Enable this Filter?
        (1) NO
        (2) YES

Enter Selection:  [2]? 2
Enter Filter Name [32 chars] []? Prot BPDU
Selection "Set Protocol Filter Entry" Complete
MSS1 BUS FILTER config for ELAN 'TKR_1'>
```

Figure 42. Adding a BPDU-Type Filter

6.4.4.3 Defining MAC Filters

Figure 43 on page 111 shows the MAC Address filter used to give access to PC101 to all protocols. This filter has to be defined for both source and destination MAC addresses. When defining the Source Address in token-ring networks, the first bit of the mask has to be set to zero to take account of the bit denoting the presence or absence of the Routing Information Field. In our example, the last four bytes have been masked out to show how a group of MAC addresses could be filtered rather than just defining a filter for a specific MAC address.

Note that when using *Talk 6*, as in Figure 42 on page 110, after defining a filter you also have to enable it and assign it to the exclude or include list. In the configuration program this is done in a different menu (see Figure 44., “Assigning BUS Filters to Include and Exclude List.” on page 112).

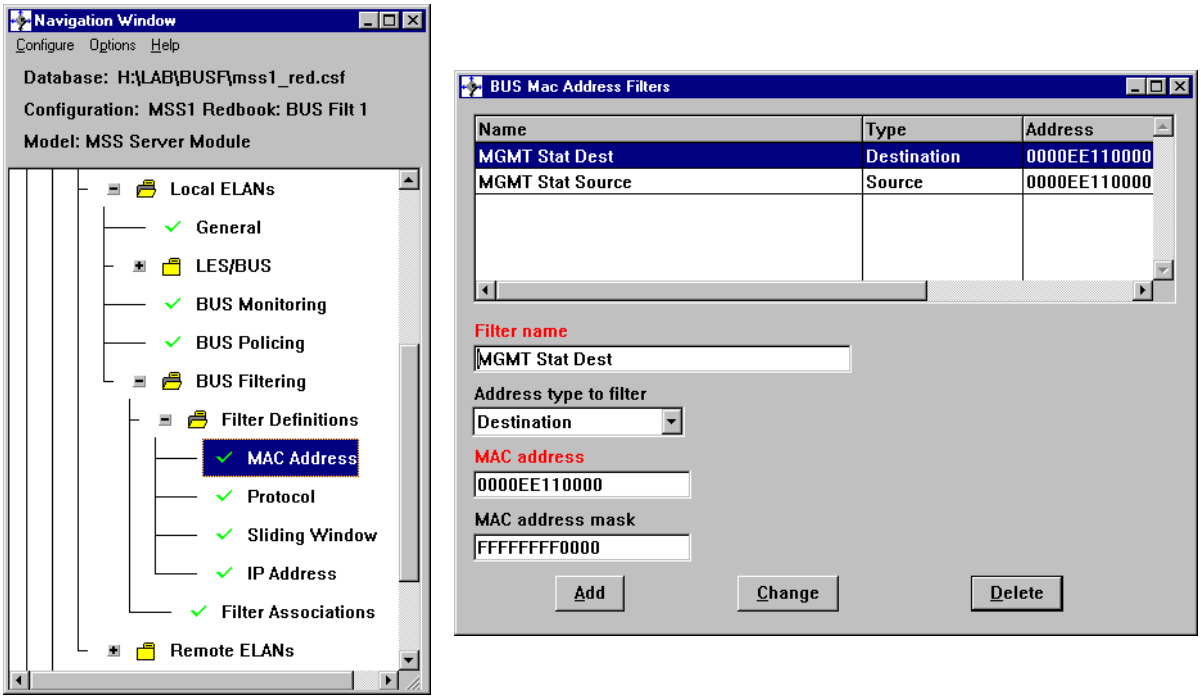


Figure 43. Adding a MAC Address Filter

6.4.4.4 Setting Global Parameters

To enable filters and to assign them to a particular LES/BUS using the configuration tool, use the *Filter Associations* menu. Select the appropriate LES/BUS and filter type and then add the created filters to the exclude or include list. See Figure 44 on page 112 for details. In our example the BPDU and IP types are assigned to the include list while NetBIOS over IP is in the exclude list. Additionally, the MAC address filters (not shown) are assigned to the include list.

Finally in the *General* part of the Filter Associations menu the default filter action and preferred list are configured. We will select the exclude list as the preferred filter list, since we want first to exclude NetBIOS over IP and only then to include IP traffic. We also set the default action to exclude in order to eliminate from the ATM network all broadcasts except the ones we have explicitly allowed. See Figure 45 on page 112 for details. In this menu it is also possible to enable/disable BUS filters at the box level.

If the command line is used instead, the BUS filter parameters are set in the BUS Filter submenu, while BUS Filtering has to be enabled at box level in the LES/BUS configuration submenu. See Figure 46 on page 113 for details.

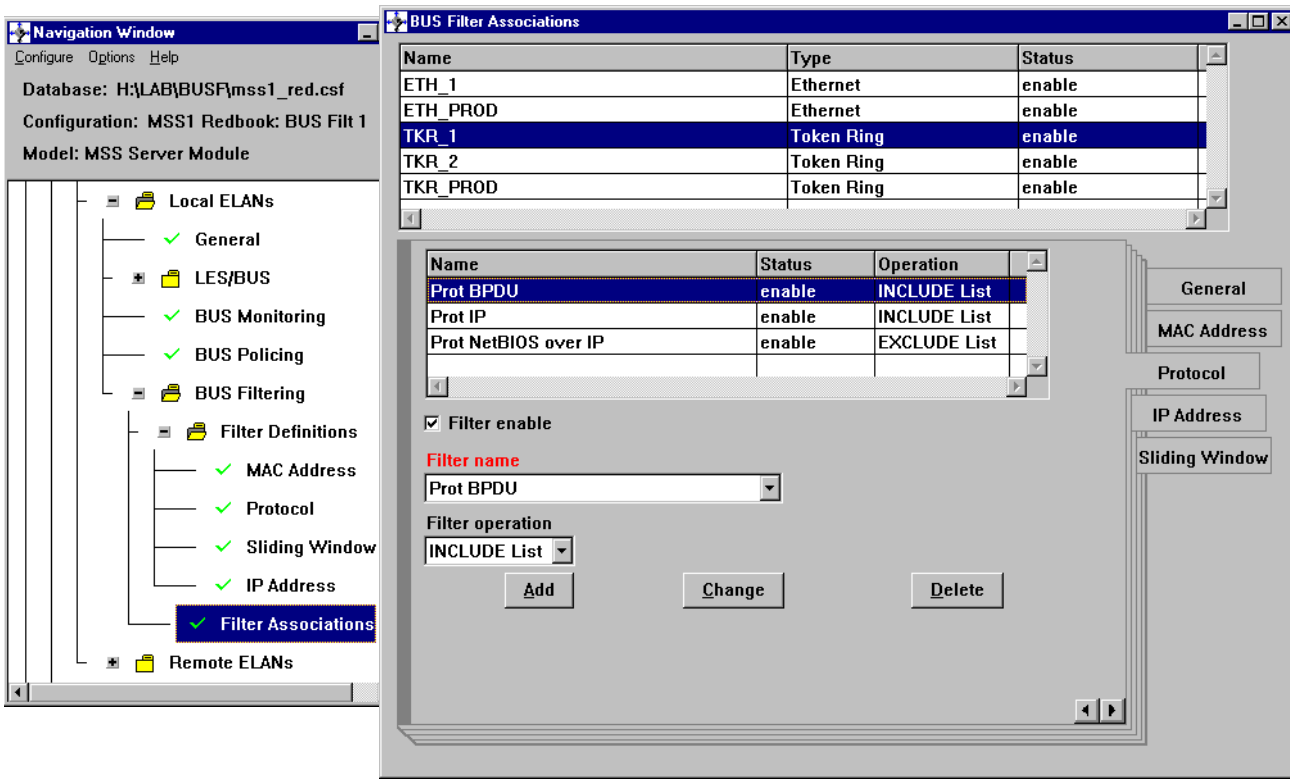


Figure 44. Assigning BUS Filters to Include and Exclude List.

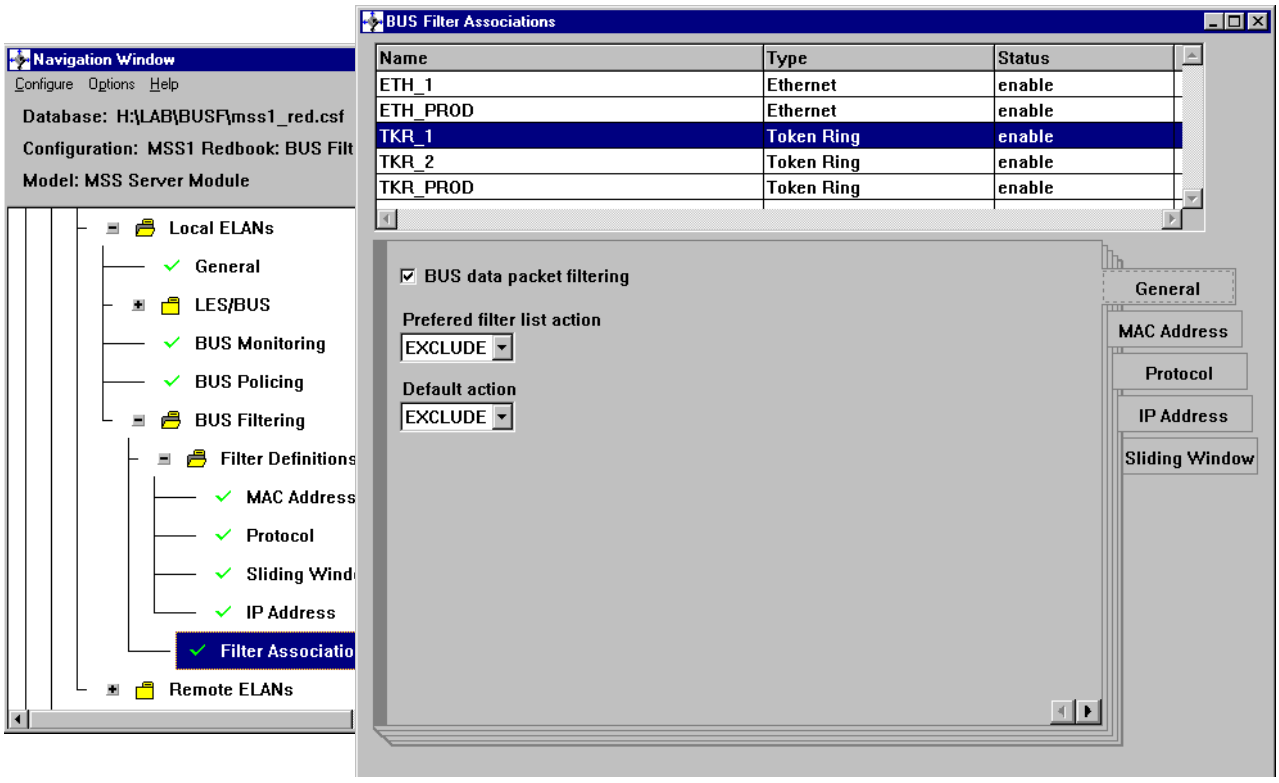


Figure 45. Configuring BUS Filter General Parameters

```
MSS1 LES-BUS config for ELAN 'TKR_1'>ENABLE BUS-FILTER
MSS1 LES-BUS config for ELAN 'TKR_1'>BUS-FILTER
BUS FILTER configuration
MSS1 BUS FILTER config for ELAN 'TKR_1'>SET DEFAULT
Select Default Action
        (1) EXCLUDE
        (2) INCLUDE

Enter Selection:  [1]? 1
Selection "Set Default Action" Complete
MSS1 BUS FILTER config for ELAN 'TKR_1'>SET PREFERRED
Select List Preference
        (1) EXCLUDE
        (2) INCLUDE

Enter Selection:  [1]? 1
Selection "Set List Preference" Complete
MSS1 BUS FILTER config for ELAN 'TKR_1'>
```

Figure 46. Setting Global BUS Filter Parameters in Talk 6

6.4.4.5 Checking BUS Filter

To check if BUS filtering is working, several actions can be performed. The first simple action is to check the active configuration and number of *filter hits*. Each time a frame is received in the BUS that matches a defined filter, a filter hit is recorded in the filter statistics before the configured action is performed. To check the active configuration use the `SHOW` command in *Talk 5*, in the BUS Filter submenu. See Figure 47 on page 114 for an example of this command, which shows the filters that we have configured as well as the number of filter hits.

The Event Logging System (ELS) function can also be used to see how the filter is working as frames are being received and passed/dropped in the LES/BUS. See Figure 48 on page 115 for an output example of the ELS for BUS filtering messages. To display the messages that apply to BUS filtering, enable the LES/BUS subsystem. For more information on ELS, refer to:

- *Nways Multiprotocol Switched Services Server Configuring Protocols and Features, SC30-3819*

We have managed to eliminate all but a few specific NetBIOS broadcasts from the ATM cloud: only those to/from allowed MAC addresses remain. We have also filtered out all other protocols except for IP.

Even a relatively simple set of filters such as this still has complex side effects which must be considered carefully before implementing filters in the first place. In our example, PC101 is the only device whose NetBIOS broadcasts are being allowed by the BUS, which means that all the other devices in the network will be able to “see” PC101. In a Windows environment, this means that PC101 will show up in the “Network Neighborhood” of all the other PCs, and that other PCs won’t because the regular broadcasts advertising PC101 will be the only ones passed by the BUS filters.

Despite this, and even though, for example, PC104 does not show up in the “Network Neighborhood” display on PC101, the filters do *not* prevent PC101 from accessing other NetBIOS stations such as PC104 provided that PC101 queries them explicitly by name, for example by using the `net use` command. In this case,

PC101 will issue a NetBIOS Name Query as a broadcast requesting the MAC address of PC104 which *will* be allowed to pass the BUS. PC104's subsequent Name Reply will not pass through the BUS but will be sent directly to PC101's MAC address, and then NetBIOS communication can proceed.

Turning this around, PC104 can *not* send a Name Query asking for PC101's MAC address if it doesn't already know it because this broadcast frame *will* be filtered by the BUS.

There's nothing wrong with all of this, but it results from the fact that BUS filters are applicable only to broadcast frames, and the way that different protocols use broadcast frames needs to be considered fully.

```
MSS1 BUS FILTER for EXISTING LES-BUS 'TKR_1'+SHOW

Bus Filter Items

Default Action:     EXCLUDE
Preferred List:     EXCLUDE LIST
      Enabled?:     YES

MAC Filter Items:

(1)Name:MGMT Stat Dest
    Address Type: DESTINATION  List: INCLUDE LIST  Enabled: YES  Hits:32
    Mac Address: 00.00.EE.11.00.00
           Mask: FF.FF.FF.FF.00.00
(2)Name:MGMT Stat Source
    Address Type: SOURCE       List: INCLUDE LIST  Enabled: YES  Hits:167
    Mac Address: 00.00.EE.11.00.00
           Mask: 7F.FF.FF.FF.00.00

 PROTOCOL Filter Items:

(1)Name:Prot NetBIOS over IP
    Protocol: NETBIOS OVER IP   List: EXCLUDE LIST  Enabled: YES  Hits:224
(2)Name:Prot BPDU
    Protocol: BPDU              List: INCLUDE LIST  Enabled: YES  Hits:376
(3)Name:Prot IP
    Protocol: IP                List: INCLUDE LIST  Enabled: YES  Hits:119

 SLIDING WINDOW Filter Items:

   NONE DEFINED

 IP Filter Items:

   NONE DEFINED
MSS1 BUS FILTER for EXISTING LES-BUS 'TKR_1'+
```

Figure 47. Checking BUS Filters

```
MSS1 ELS>DISPLAY SUBSYSTEM les all
MSS1 ELS>
MSS1 *FLUSH 2
MSS1 *TALK 2
00:16:17  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x0000EE110
000                                                                              1
00:16:17  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.          2
00:16:17  LES.407: BUSFILTER:'TKR_1':Match Exclude List. cmp value = 0xAAAA03000
0000800450000000000000000000000000000000000000000008A008A
00:16:17  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.          3
00:16:17  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0xAAAA03000
0000806                                                                          4
00:16:17  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x0000EE110
000
00:16:17  LES.407: BUSFILTER:'TKR_1':Match Exclude List. cmp value = 0xAAAA03000
000080045000000000000000000000000000000000000000008B                            5
00:16:18  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.
00:16:18  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.
00:16:18  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x424203          6
00:16:20  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x424203
00:16:22  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x424203
00:16:24  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x424203
00:16:25  LES.407: BUSFILTER:'TKR_1':Match Exclude List. cmp value = 0xAAAA03000
000080045000000000000000000000000000000000000000008B
00:16:26  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x424203
00:16:27  LES.408: BUSFILTER:'TKR_1':Match Include List. cmp value = 0x0000EE110
000
00:16:27  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.
00:16:27  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.
00:16:27  LES.409: BUSFILTER:'TKR_1':No Match. Dflt performed-EXCLUDE.
```

Figure 48. ELS for BUS Filters

Notes:

1 A frame with a source or destination MAC address equal to the one defined in the MAC address filter has arrived. The frame has been passed since the filter is in the include list.

2 A frame arrives that does not match any filter. The default action is performed.

3 A frame has been dropped because it matches a filter defined in the exclude list. It corresponds to a NetBIOS over IP frame.

4 An IP frame has been passed because it is in the include list.

5 Another NetBIOS over IP frame.

6 These frames are BPDUs. They are passed because a filter for them is defined in the include list.

6.4.4.6 Setting BUS Filters to Obtain Security in an IP Environment

In our second example, our goal is to provide a network in which IP subnet 8.8.4.x cannot be reached from an outside subnet unless the station trying to reach it has IP address 8.8.1.101. Every other protocol is allowed.

This example would match a typical customer network with some security needs in which some IP subnets should never be accessible from the outside, except

perhaps for certain administrative stations with a given MAC address or IP address.

The scenario will be the one depicted in Figure 49 on page 116. In this case reasonable planning of IP address and certain IP routers could possibly have led to the same result, perhaps with even better security, but this is not always possible. Many networks are initially designed without taking security aspects into consideration, and then users and IP addresses are added or IP networks are split into subnetworks. In other networks the users are simply dispersed geographically and cannot be joined together in the same IP subnet.

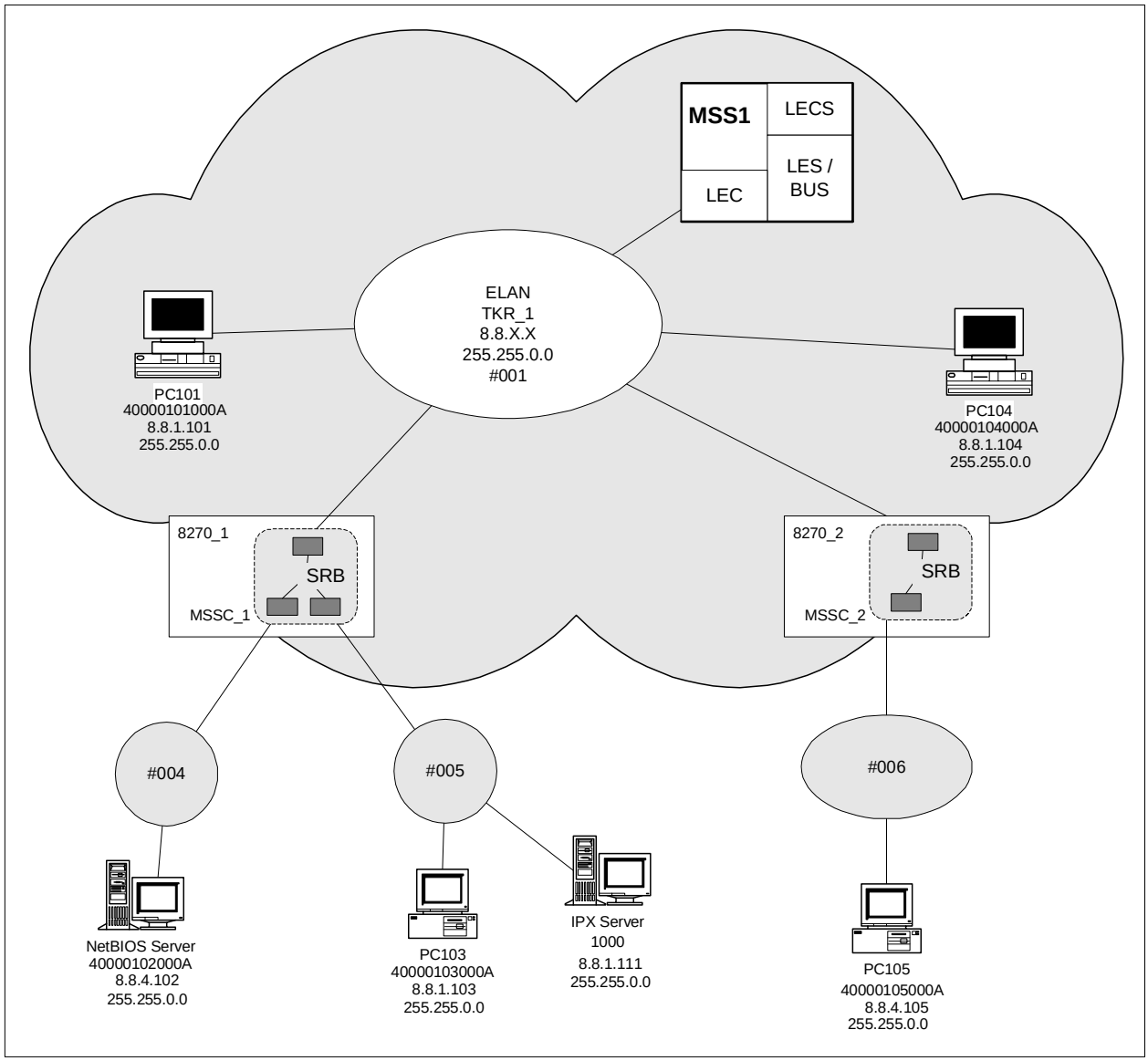


Figure 49. Network Diagram for BUS Filter Example 2

What we have tried to do in this example is to show a flat IP Class A network with a Class B subnet mask, 255.255.0.0. There are no routers in this network, because the MSS Clients at the edge of the network are acting as source-routing bridges. We want to provide some security for a subset of IP devices in this

network, those which can be considered to be in the Class C subnet 8.8.4.0. BUS filters allow some security in the sense that they can prevent ARP requests from outside this subnet from flowing. Remember that since we are only dealing with broadcast frames here, ARP is the only sort of IP frame that we can control with filters.

By adding filters we prevent any IP station from outside the 8.8.4.0 subnet from discovering the MAC address of any station inside the subnet. But note that if an IP station inside the 8.8.4.0 subnet wants to discover the MAC address of an IP station outside the 8.8.4.0 subnet, the ARP frame will flow, and communication will proceed normally. In this case, the target IP station will cache the MAC address of the source station in its ARP cache and will subsequently be able to use the entry to initiate communication itself. So there is a way for stations outside the “private” subnet to talk to devices inside the subnet, but only if communication is initially started from within the “private” subnetwork.

Remember that all the IP stations are still on the same ELAN, connected to the same LES, so there is no way that BUS filters can prevent communication between IP stations on the same ELAN if the endstations know the MAC addresses of the IP partners with which they want to communicate. In this example, BUS filters could be considered a security enhancement but certainly not a complete security measure.

In order to configure this network environment, the following filters have to be defined:

1. **Include** access for Management IP Address (or MAC Address). Only needed for Source addresses. This allows some management stations to overcome the filters.
2. **Include** access for Source IP Addresses in the private IP subnet. This is needed only if stations in the secured IP subnet need to talk to stations outside.
3. **Exclude** access for any frame with a destination IP address in the secured IP subnet. This ensures that devices which are not in the “private” IP subnet, and which therefore do not pass the previous include filter, are unable to send ARP frames into the subnet.
4. **Include** default access for everything else.

The preferred list has to be chosen as the include list to ensure that we check the first two points initially. Then, according to point 3, any IP broadcast frame destined for the “secured” IP subnet from outside and not coming from our designated management station will be filtered.

Defining IP Address Filters

To set an IP Address filter using the config tool, the procedure is similar to the one we used before for MAC address or protocol filters. This time, the IP Address submenu is to be selected (see Figure 50 on page 118).

The data that has to be entered is: Filter Name, Address Type (source or destination), subnet address and subnet mask. We define three filters: one for the source IP address of the “management” station and two for the secured IP subnet (source and destination).

If the command line is used some more data has to be entered at the same time as defining the filters since the filters are also assigned at this time to the exclude or include list, and are enabled or disabled (see Figure 51 on page 119).

To assign filters to the exclude or include list and enable them using the config tool, the same procedure as for the previous example is used. In this case, the preferred list should be the include list, and also, the default action should be to include. When assigning filters, set them as shown in Figure 52 on page 119.

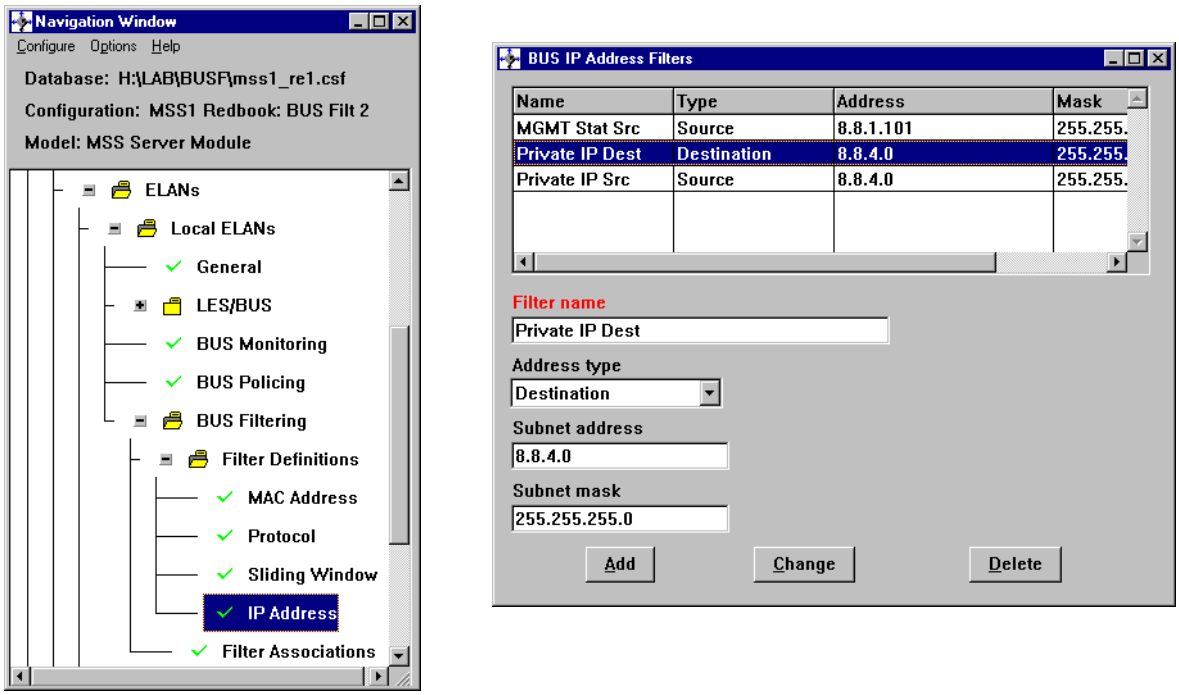


Figure 50. Defining IP Address Filters with Config Tool

```
MSS1 LES-BUS config for ELAN 'TKR_1'>BUS-FILTER
BUS FILTER configuration
MSS1 BUS FILTER config for ELAN 'TKR_1'>ADD IP
Select IP address type
        (1) SOURCE IP ADDRESS
        (2) DESTINATION IP ADDRESS

Enter Selection:  [1]? 1
Enter IP address or subnet [0.0.0.0]? 8.8.1.101
Enter IP subnet mask [0.0.0.0]? 255.255.255.255
Select Filter List
        (1) EXCLUDE
        (2) INCLUDE

Enter Selection:  [1]? 2
Enable this Filter?
        (1) NO
        (2) YES

Enter Selection:  [2]? 2
Enter Filter Name [32 chars] []? MGMT Stat Src
Selection "Set IP Filter Entry" Complete
MSS1 BUS FILTER config for ELAN 'TKR_1'>
```

Figure 51. Adding an IP Address Filter from the Command Line

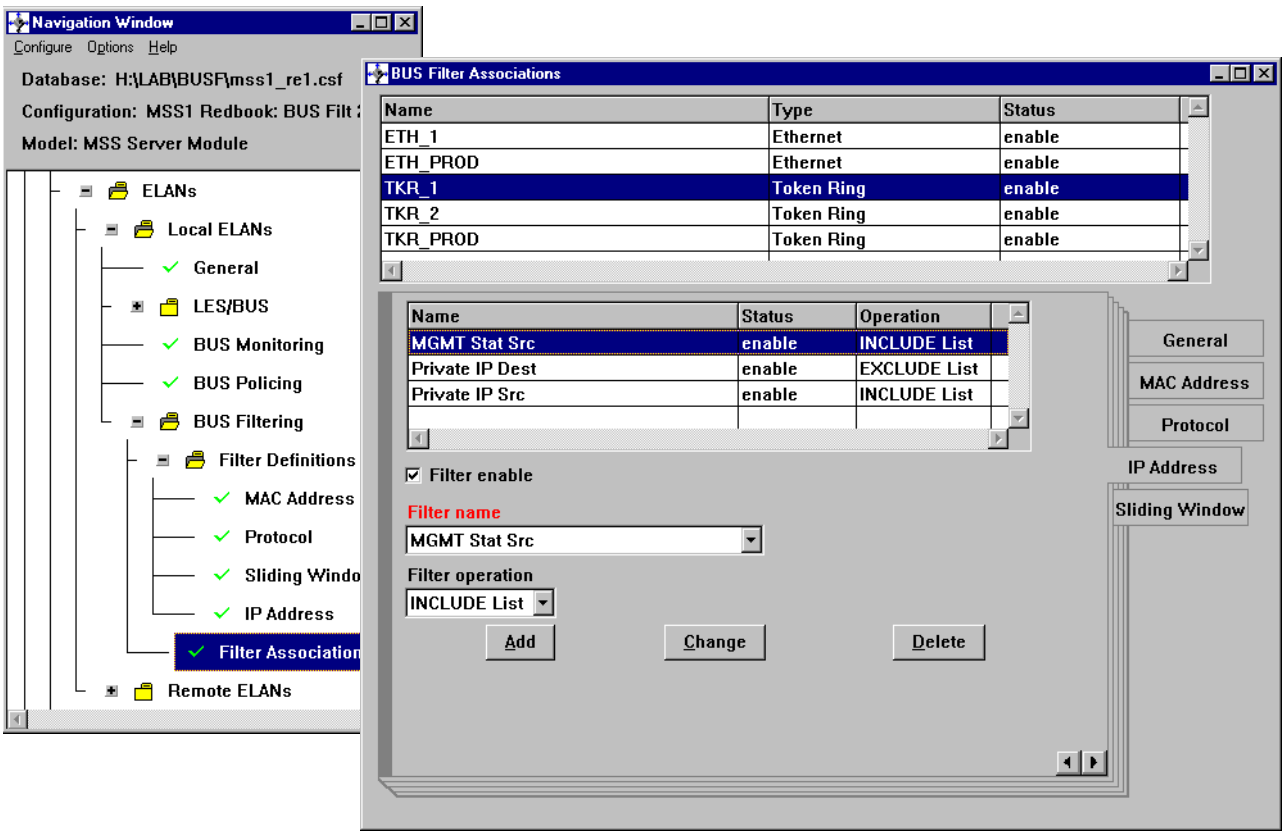


Figure 52. Assigning IP Address BUS Filters to Include and Exclude Lists

6.5 BUS Police

In the following section we will describe the BUS Police function, which is one of the new enhancements to the BUS in MSS Server Release 2.2. A brief description of it will be given along with a description of how to configure it.

BUS Police gives some protection from broadcast storms, if they come either from a malfunctioning device or from an intentional external attack to the network.

BUS Police uses BUS Monitor (shipped with MSS Server R1.0) and BUS filtering (shipped with MSS Server R2.2) to provide dynamic traffic management. BUS traffic is managed by sampling BUS data traffic and dynamically installing BUS MAC filters for devices that exceed a pre-configured BUS usage threshold. This type of traffic management can protect ATM LANE backbone networks from intentional sabotage or from unintentional misbehaving devices.

6.5.1 Introduction

If a network adapter is malfunctioning it could send a storm of broadcast frames into the network. In ATM networks this would mean that the BUS will soon be overloaded and all network traffic would be stopped. BUS Police gives a certain amount of protection in this case. It can also work when the broadcast storm is intentional, such as in a deliberate attempt to sabotage the network.

In both cases, the BUS will see an increasing amount of multicast and broadcast traffic coming from a particular MAC address. The BUS Monitor function detects this situation and BUS Filters stop it from continuing. So BUS Police is really a way of automating the detection and correction process. Without it, human detection, analysis and correction is going to be required which could result in the network being unavailable or degraded for a considerable time until the problem is fixed.

Call Pacing - A Related Issue

A situation can also arise in which an ATM-attached device tries to set up calls continuously. In this case the overhead in the network is due not to broadcast traffic, but rather to *signalling* traffic. BUS Police is not effective here simply because signalling does not go through the BUS. But signalling storms also affect the switches and slow down the overall network. They could also slow down the MSS itself if all Call Setups are directed toward it.

As stated, BUS Police will not provide protection in this situation but *Call Pacing* could. Call Pacing is a feature introduced in the 8265 switches in Release 4. It is useful when an end node receives too many Call Setups at a time to be able to process them (ATM switches can typically process and forward many more Call Setups per second than end nodes can handle). In this case, Call Pacing will help by placing the Call Setups in a queue and space them when re-sending them to the end node.

For more information about Call Pacing see *8265 Nways ATM Switch User's Guide,* SA33-0456.

6.5.2 BUS Police Implementation

BUS Police relies on BUS Monitor and BUS Filtering to work. Therefore BUS Police can only be enabled when the BUS is set to Adapter or System mode. VCC-Splice mode is not supported. BUS Monitor needs to be enabled prior to configuring BUS Police. BUS Filter does not need to be enabled explicitly; the BUS Monitor function will itself set up filters and use them even if BUS filtering is disabled in the MSS Server.

BUS Monitor was shipped with MSS Server Release 1.0. Its functions remain the same in Release 2.2, except that the *Time between samples* parameter is now defined in seconds, rather than in minutes. This change is needed to accommodate the needs of the BUS Police function, since a sample time of several minutes would not be suitable for good protection from intermittent broadcast storms.

BUS Police uses BUS Monitor to detect when devices are over-using the BUS. At the end of each BUS Monitor sample period, BUS Police checks the BUS Monitor results to determine if any device in the list has exceeded the BUS Police Usage Threshold. All devices exceeding the BUS Police Usage Threshold are immediately filtered using BUS MAC Address Filters.

BUS Police provides two options for the implementation of these dynamically-created filters:

- *Temporary*, in which a filtered MAC address will be removed from the BUS Police Filter List after the next BUS Monitor sample period if it no longer exceeds the BUS Police usage threshold.
- *Permanent*, in which the filtered MAC address will never be removed from the BUS Police Filter List even if the transmit rate of the offending MAC address subsequently falls below the BUS Police usage threshold. MAC filters installed by the BUS Police in this mode can only be removed through manual intervention.

BUS Police Filters are implemented in a similar manner to static BUS MAC Address Filters; BUS Police Filters are not linked to the BUS filter's exclude and include Filter Lists but are instead maintained on a separate BUS Police Filter List. This allows the BUS Police function to operate even if BUS Filtering is disabled. Another difference is that filters on the BUS Police Filter Lists are applied *after* BUS Monitor has processed the frame whereas static BUS Filters are applied *before* BUS Monitor processes the frame. This allows BUS Monitor to continue statistical monitoring of MAC addresses even for MAC addresses that are actually being policed (that is, currently being prevented from using the BUS).

For each device that is being policed, a Source MAC Address BUS Filter is inserted in the BUS Police exclude Filter List. Devices that are granted BUS Police *immunity* will already have a corresponding Source MAC Address BUS Pass Filter inserted in the BUS Police include Filter List. The order in which BUS Police filters are processed is not significant, since the MAC addresses on each list are mutually exclusive. In other words, if a device has been granted immunity and therefore has its MAC address on the include list, BUS Police will never put its address on the exclude list even if it exceeds the Usage Threshold. A frame discard counter is maintained for each BUS Police Filter and may be viewed from the console.

BUS Police *Immunity*, mentioned above, can be granted to specific devices which are allowed to transmit data across the BUS at any rate regardless of the configured BUS Usage Threshold. These devices are still monitored by the BUS Monitor and their MAC addresses will still be listed in the BUS Monitor results if they are on the Top *N* users of the BUS, since BUS Monitor function is independent from the BUS Police.

6.5.2.1 Limitation

BUS Police has an apparent limitation. During each BUS Monitor sampling period only the first 1,000 unique MAC addresses detected can be policed. If there are more that 1,000 devices sending traffic through the BUS during any one sample period, some devices could exceed the BUS usage threshold and not be caught. However, since the 1,000 entries of the BUS Monitor MAC tracking table are reset at the beginning of each new sampling period, it is actually extremely unlikely that any heavy user of the BUS could escape detection across multiple BUS Monitor sampling intervals, even in very large ELANs with thousands of LAN Emulation Clients.

6.5.3 Configuring BUS Police Parameters

Prior to enabling BUS Police, the BUS Monitor function has to be enabled and configured. The default BUS Monitor settings are not optimal for the BUS Police function, since they were originally designed just to gain statistical data about BUS usage.

BUS Police and BUS Monitor are configured independently for each LES/BUS defined in the MSS Server. The configuration methods supported are the same as for the BUS Data Frame Filters:

- Non-dynamic configuration using Configuration Program or *Talk 6*. The MSS Server needs to be rebooted to activate a new configuration. Changes are permanent.
- Dynamic configuration using *Talk 5* via the Telnet or Web interface. BUS Police and BUS Monitor can be enabled, modified, and disabled from the MSS Server console while the ELAN is operational. Any changes to the BUS Police or to BUS Monitor made from *Talk 5* are lost when the MSS Server is rebooted or if the ELAN is restarted.

The steps required to enable and configure BUS Monitor are:

1. Configure LES/BUS properly for a given ELAN in the usual way.
2. Enable BUS Monitor and configure it with the optimum values for BUS Police.
3. Enable BUS Police and set the police parameters.
4. Add devices to be granted BUS Police immunity (if needed).

6.5.3.1 Configuring BUS Monitor for BUS Police

Default BUS Monitor settings are not optimal for BUS policing. The *Effective Sampling Rate* is set too low to enable effective protection. The effective sampling rate is the rate at which the BUS Monitor actually samples frames, measured as the percentage of the number of frames sampled out of the total of frames. For effective protection while not overburdening the BUS, an effective sampling rate of 20% should be chosen (the default effective sampling rate is

0.05% which is a good value if BUS Police is not enabled and we are only interested in getting statistics about BUS usage).

The recommended BUS Monitor parameter settings are:

Table 8. Recommended BUS Monitor Settings

Top MAC addresses to record:	10
Duration of samples (sec):	179
Time between samples (sec):	180
Sampling rate:	5

With these numbers the effective sampling rate is 19.89%: to calculate the effective sampling rate, use the following formula:

$$\text{Effective Sampling Rate (\%)} = \frac{\text{Duration of Samples (sec)} \,/\, \text{Time between Samples (sec)}}{\text{Sampling Rate}} \times 100$$

Figure 53. Calculation of Effective Sampling Rate

The “Time between samples” is actually the time between the *start* of the samples, so in this case there is in fact only one second between the end of one sample and the start of the next sample. The sampling rate is the fraction of all the frames received that the sample consists of. What we are doing here is to make the BUS Monitor analyse samples every three minutes; the samples themselves cover one second less than three minutes in duration and consist of a fifth of the frames actually received during this period.

Recommended settings can be changed to suit a particular need. Reducing the time between intervals will provide greater protection and less likelihood of missing a specific MAC address but will also increase the CPU overhead. However we recommend modification of these values only slightly to keep the effective sampling rate to values near 20%. If this value is still too high for your network because of heavy broadcast activity, then we suggest using a lower value around 15% or even 10% but keeping the ratio D*uration of Samples/Time between Samples* near to 1.

If time between samples is increased and the sampling rate is simultaneously decreased in the same proportion then the effective sampling rate will remain the same (assuming that the duration of the sample is not changed) and the effect will be the following:

- The MSS will be counting more frames than before (sampling rate decreased) during the duration of the sampling, therefore the CPU overhead in this period will increase. As more frames are checked, there is an increase in security in this phase and potentially fewer MAC addresses could escape being filtered.
- The remainder of the period sample (time between samples - duration of samples) will increase and during this phase the MSS will not be counting any frame, so the CPU overhead is minimal. As no frame count is performed during this phase, the BUS Police filters remain unchanged.
- The result is an increasingly asymmetrical use of the CPU.

6.5.3.2 Configuring BUS Police

Before enabling BUS Police a LES/BUS has to be created and BUS Monitor has to be enabled and configured. Using the command line, the command ENABLE BUS-POLICE then activates the BUS Police and gives access to the BUS Police configuration menus:

```
MSS1 Config>NETWORK 0
ATM user configuration
MSS1 ATM Config>LE-SERVICES
LAN Emulation Services user configuration
MSS1 LE Services config>LES-BUS 2
LES-BUS configuration
MSS1 LES-BUS config for ELAN 'ETH_1'>ENABLE BUS-POLICE                  1
MSS1 LES-BUS config for ELAN 'ETH_1'>
MSS1 LES-BUS config for ELAN 'ETH_1'>BUS-POLICE
BUS POLICE configuration
MSS1 BUS POLICE config for ELAN 'ETH_1'>?
Possible completions:
        ADD immunity list item
        REMOVE immunity list item
 ...    SET bus police parameters
        SHOW immunity list items
        EXIT
(you may cycle through these commands by pressing the TAB key)
MSS1 BUS POLICE config for ELAN 'ETH_1'>SET DURATION                    2
Select Bus Police filter Duration
        (1) TEMPORARY
        (2) PERMANENT

Enter Selection:  [1]? 1
Selection "Enter Bus Police filter Duration" Complete
MSS1 BUS POLICE config for ELAN 'ETH_1'>SET THRESHOLD 100               3
Selection "Enter Bus Police filter threshold" Complete
MSS1 BUS POLICE config for ELAN 'ETH_1'>
```

Figure 54. BUS Police Configuration Commands

Notes:

1 Enables BUS Police. By default BUS Police is disabled.

2 Specifies the installed BUS Police MAC filter duration. The default value is temporary (1):

1. TEMPORARY: Any MAC filter installed by BUS Police is automatically removed if the offending MAC address's BUS transmit rate subsequently falls below the BUS Police Threshold.
2. PERMANENT: Instructs BUS Police never to remove any installed MAC filters even if the offending MAC address's BUS transmit rate later falls below the BUS Police Threshold. MAC filters installed by BUS Police in this situation can be removed only through manual intervention.

3 Sets the BUS Police BUS transmit threshold in packets per second. Packets originating from MAC addresses that exceed this rate during a BUS Monitor Sample Interval will be filtered. The default value is 50.

If the configuration program is used instead of the command line or the Web interface (which is really just a pretty version of the command-line interface), the screen will look like Figure 55, and the same parameters apply.

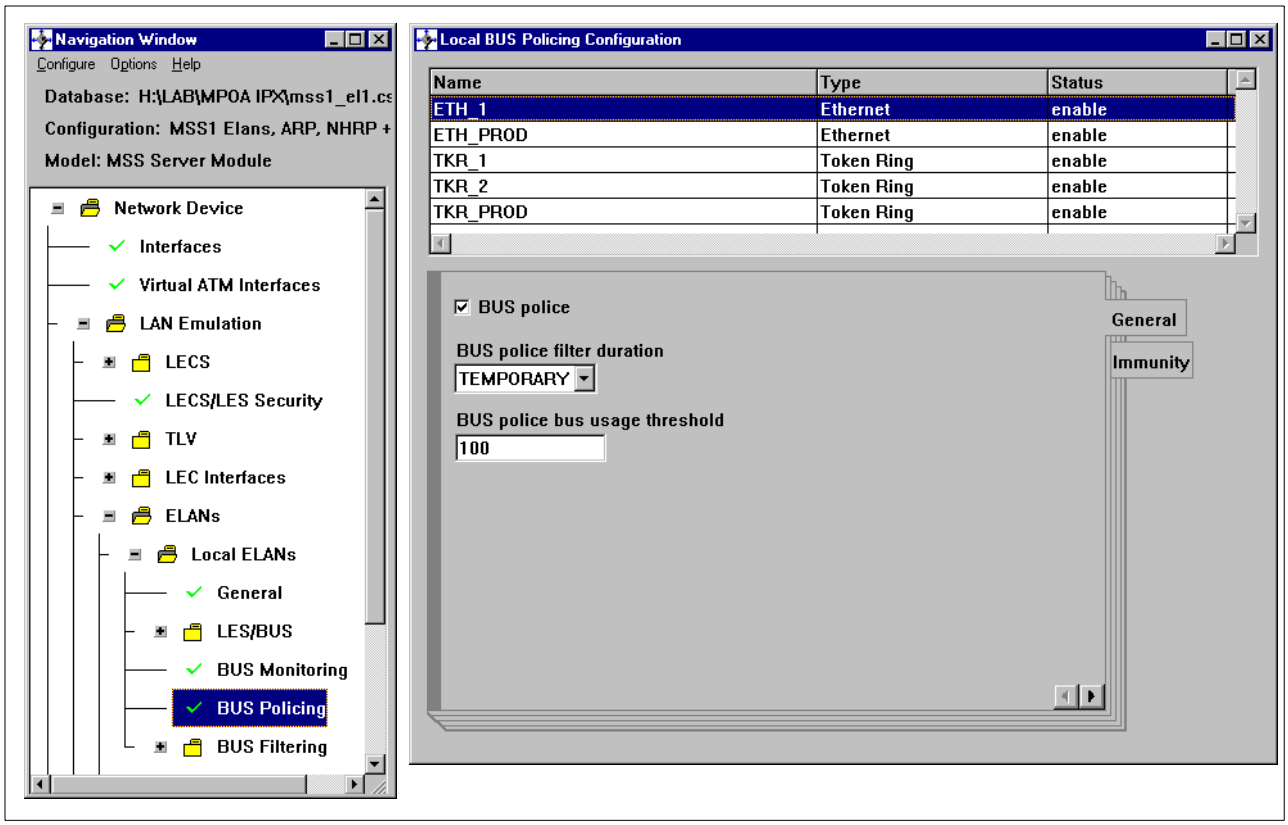


Figure 55. Configuring BUS Police with the Config Tool

6.5.3.3 Granting BUS Police Immunity

Some devices need to be granted immunity from BUS policing: devices such as routers or SNMP network management stations may need to be exempted from BUS policing. Devices that are granted immunity from BUS policing may transmit data across the BUS at any rate regardless of the configured BUS Usage Threshold.

Granting BUS Police immunity is configured by MAC Address. Multiple MAC Addresses may be exempted from BUS policing. The only limitation is the amount of SRAM needed to store the MAC Address information. All definitions only affect BUS Police in a particular LES/BUS instance. If BUS Police is configured for multiple BUS instances, then Police immunity lists (if needed) have to be defined for each BUS.

BUS Police immunity can be granted from *Talk 6* or *Talk 5* menus. Use the command `ADD` in the BUS Police menu, as shown in Figure 56. It is also possible to use the config tool. Immunity granted in *Talk 5* takes place immediately and there is no need to reset the MSS, but all definitions are lost at the next reset.

```
MSS1 BUS POLICE for EXISTING LES-BUS 'ETH_1'+ADD
Enter Source MAC address to be granted immunity: [FF.FF.FF.FF.FF.FF]?
00.04.AC.FF.C9.FF
Selection "Add Bus Police Mac Address" Complete
MSS1 BUS POLICE for EXISTING LES-BUS 'ETH_1'+
```

Figure 56. Granting BUS Police Immunity

6.5.3.4 Monitoring BUS Police

To monitor BUS Police use the command `SHOW filter lists` in *Talk 5* in the BUS Police submenu. This shows the actual configuration of BUS Police and a list of devices granted immunity along with the list of BUS Police filters in use and the number of hits per MAC address. An example of this command is shown in Figure 57, using the Web interface.

To get more information it is also very useful to check the Top N users of the BUS in the BUS Monitor statistics, using the *Talk 5* command `STATISTICS DISPLAY BUS RATE`. The output of this command is shown in Figure 58 on page 127.

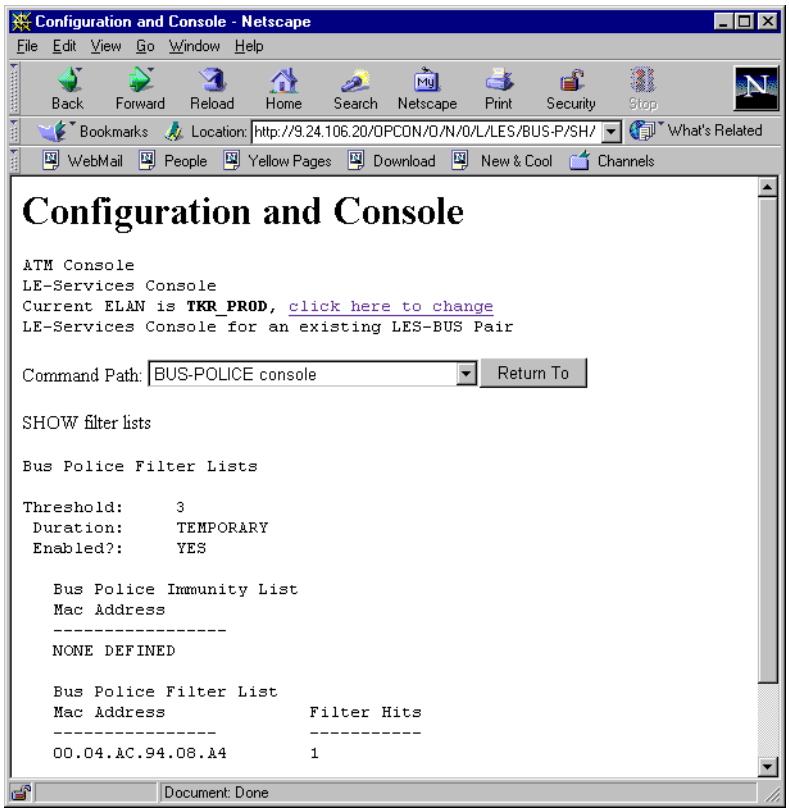


Figure 57. Monitoring BUS Police

```
MSS1 EXISTING LES-BUS 'TKR_PROD'+STATISTICS DISPLAY BUS RATE
   -BUS Monitor Status-
   Currently in a sample interval ?            yes
   Sample interval scheduled to stop in:       2 minute(s), 50 second(s)
   Next sample interval scheduled in:          2 minute(s), 51 second(s)

   -Results of Last Complete Sample-
   BUS Monitor sample interval started at:     01.57.17.00   (System Up Time)
   Duration of sample interval:                179 second(s)
   # Top Hosts Actually Recorded:              10
   # Frames Received in sample interval:       49701
   # Frames Sampled in sample interval:        9941
   Frame sampling rate:                        1 out of 5
                                                                    Frame Rate
Rank  Source MAC Addr.    Associated LEC ATM Address                (pkts/sec)
----  ----------------    ---------------------------------------   ----------
   1  00.04.AC.FF.C9.FF   39999999999999000099990101420082720000 80          6   1
   2  00.06.29.6B.23.2A   39999999999999000099990101420082720000 80          3
   3  08.00.5A.22.64.17   39999999999999000099990101420082720000 80          2
   4  00.06.29.84.1C.B5   39999999999999000099990101420082720000 80          2
   5  00.06.29.6A.AD.EF   39999999999999000099990101420082720000 80          2
   6  10.00.5A.89.39.25   39999999999999000099990101420082720000 80          2
   7  00.20.35.2A.E1.B9   39999999999999000099990101420082720000 80          1
   8  10.00.5A.8A.75.D6   39999999999999000099990101420082720000 80          1
   9  08.00.5A.CD.DE.D0   39999999999999000099990101420082720000 80          1
  10  00.04.AC.60.36.D5   39999999999999000099990101420082720000 80          1
MSS1 EXISTING LES-BUS 'TKR_PROD'+
```

Figure 58. Top Ten Users of the BUS

Notes:

1 This MAC address has exceeded the BUS Police threshold and therefore appears in Figure 57 on page 126 as filtered.

The BUS Police threshold is configured to *4 packets per second* in our example; we had to do this for our small test network in order to show addresses actually being filtered. This value is unrealistically low for a real network; the default value is 50.

6.5.4 Conclusions

BUS Police is used in conjunction with BUS Filters and BUS Monitor to control the rate of broadcast frames that a given station can send to BUS. If this rate exceeds a pre-configured value, then a MAC address filter is placed in the BUS and this device is no longer allowed to send any more broadcast frames, thus providing protection from malfunctioning devices and also from intentional attacks and broadcast storms.

Care has to be taken when configuring BUS Monitor so that it does not cause excessive overhead in the CPU and that parameters are set appropriately to achieve the desired protection.

6.6 IP Multicast VLANs

IP Multicast VLAN is another approach to reducing the broadcast overhead in the network. It does so by reducing unnecessary forwarding of IP multicast data frames on the ASRT bridge or on the SuperELAN bridge by ensuring that these frames are sent only to those devices which will be interested in receiving them.

This function expands the Dynamic Protocol Filter (DPF) VLAN capabilities introduced in MSS Release 1.1. Previously, MSS R2.0.1 and R2.1 had expanded the DPF capabilities by adding more types of VLANs that could be created. In MSS Release 2.2, IP Multicast VLAN support is added.

DPF (and IP Multicast VLANs as a subset of DPF) and BUS Filters try to solve the same problem but put the effort in different entities in our network. BUS Filters work like broadcast manager (BCM) at the BUS level, and try to cut the broadcast before it is sent into the ELAN. On the other hand, DPF works like bridging broadcast manager (BBCM) at a bridge level, trying not to spread broadcasts indiscriminately that originate in one ELAN to other ELANs connected to the same bridge.

IP Multicast VLANs restrict IP multicast data to ports with stations in the same IP multicast group. IP Multicast VLANs may be configured manually (by supplying an IP Group address) or created dynamically based on IGMP packets which appear on the various ports of the MSS. IGMP report frames are used to determine inclusion in the forwarding domain for a particular IP Multicast VLAN. IP Multicast VLANs only filter IP Multicast Data frames.

To ensure that an IP Multicast VLAN does not interfere with the IGMP protocol, IGMP frames are not filtered. DPF “snoops” on the IGMP report frames to identify new IP multicast groups, and may create a new VLAN when a new IP multicast address is detected. No user configuration is required when automatic creation is enabled.

In this section we will review the IP Multicast VLANs support in the MSS 2.2. This function was already announced for the MSS Release 2.1 but was finally left out of the code.

6.6.1 Introduction

Before delving deeply into IP Multicast VLANs, let’s review some aspects of DPF and IP Multicast frames in general.

6.6.1.1 Dynamic Protocol Filtering Basics

Dynamic Protocol Filtering is a method by which a bridged network may be partitioned into several protocol-specific Virtual LANs. Its purpose is to limit the scope of frames that are normally forwarded over all active spanning tree ports. It behaves just like regular filters but has the advantage that DPF dynamically turns filters on and off on each bridge port based upon the traffic seen from this port. The bridged network can thus be dynamically partitioned into protocol-specific subnetworks. The partitioned domains, or VLANs, can overlap.

Dynamic Protocol Filtering monitors traffic over each bridge port, learning the protocols and subnets that are being used on that port. The user only needs to enable DPF for a particular (or several) protocols and subnets, and define which subnets will be present in the network. For example, for IP VLANs, the user could specify that IP subnets 12.0.0.0 and 192.0.0.0 exist but the user doesn’t need to assign subnets to individual ports because DPF does this job dynamically, as its name implies.

For each configured subnet, the subset of bridge ports on which traffic for that subnet is being received is referred to as the *Forwarding Domain* of that subnet. DPF manages the forwarding domains for each subnet. Broadcast and multicast

frames for a particular subnet will not be forwarded on bridge ports that are not in the forwarding domain of that subnet.

DPF and BBCM each attack the same problem of unnecessary propagation of certain multicast and broadcast frames over bridge ports. BBCM operates by transforming broadcast frames into unicast frames. DPF operates by identifying which ports are active in which subnets and forwarding traffic for those subnets only on those ports. BCM and BUS Filters also attack the same problem, but they work at a BUS level rather than a bridge port level. See 6.7, “Comparison Between BCM, BBCM, DPF VLANs and BUS Filters” on page 132 for a detailed comparison between these functions.

For the initial release of DPF, in MSS Server Release 1.1, the supported protocols were IP, IPX and NetBIOS. In MSS Server Release 2.0.1, DPF VLANs were extended to include sliding-window VLANs and MAC addresses VLANs. In MSS Server Release 2.1, DPF VLANs were further extended to include Port-based VLANs, and in MSS Server Release 2.2 support for IP Multicast VLANs has now been added.

This book covers only IP Multicast VLANs. To know more about other types of VLANs and DPF the reader should consult earlier MSS documentation such as:

- *Understanding and Using MSS Server Release 1.1 and 2.0,* SG24-2115
- *MSS Release 2.1, Including the MSS Client and Domain Client,* SG24-5231

The user can manually include or exclude any bridge port in any VLAN, overriding dynamic configuration. This is useful when ’quiet’ devices (such as printers) are attached to a port and the MSS is unaware of their existence owing to the lack of regular broadcasts originating from the device.

6.6.1.2 IP Multicast

Multicasting is the simultaneous transmission of a single data element to multiple destinations. IP multicasting is a Layer 3 protocol based on RFC 1112[1] and is used for transmitting IP datagrams from one IP source to many IP destinations in a local or wide area network. It is the Internet abstraction of hardware multicasting. IP multicasting and broadcasting have a number of uses, but the most well-known use is for distributing audio and video. IP multicast and broadcasting are also used by major routing protocols (RIP, OSPF, BGP etc.) and networking services such as BOOTP and Data Link Switching (DLSw).

By adding IP Multicast VLANs support to MSS DPF capabilities, IP Multicast traffic can be restricted to the bridge ports with stations in a single IP Multicast group.

6.6.2 IP Multicast VLANs Implementation

IP Multicast VLANs reduce unnecessary forwarding of IP multicast data frames onto the network by forwarding this kind of frame only to ports with devices in the same multicast group and over ports attached to multicast routers.

When an IP multicast frame arrives at the MSS, it is passed to the DPF. The target address is compared against all known IP Multicast VLANs. If the target group IP address is known, then the frame is sent to all the ports in the forwarding

[1] S. Deering, *Host Extensions for IP Multicasting*, RFC 1112, August 1989

domain of the VLAN. If the target group IP address is not known, then it is forwarded to all active spanning tree ports.

IP multicast groups can be manually entered into the IP Multicast VLANs configuration but, unlike other VLANs, IP Multicast VLANs can be automatically created without user intervention.

*Auto-creatio*n is enabled if an IP Multicast VLAN exists for the 'all IP hosts address' (224.0.0.1) and if this VLAN is enabled. If this IP group address is present but is disabled, then auto-creation is disabled and the user must enter all the IP group addresses to use.

If the IP group address 224.0.0.1 is not present at box initialisation then one is automatically created and enabled. This means that the auto-creation feature will be activated.

Note

To disable auto-creation permanently, disable the 224.0.0.1 VLAN. Don't delete it.

DPF "snoops" on IGMP report frames to create IP Multicast VLANs automatically and also to create the forwarding domain associated to an IP Group address.

If an IGMP report frame is received it is checked it see if it matches a known IP Multicast VLAN; if so then its entry port is added to the forwarding domain of the IP Multicast VLAN if not already present.

If the frame does not match any known IP Multicast VLAN and if automatic creation is enabled, then a new IP Multicast VLAN is created with the default port configuration as specified in the definition for the 224.0.0.1 VLAN. The 224.0.0.1 VLAN contains the initial port, config, aging time and MAC address tracking status that will be applied to each auto-created VLAN.

While IGMP report frames determine the location of multicast hosts, the reception of IGMP query frames determines the location of multicast routers. Ports attached to multicast routers are included in the forwarding domain of every IP Multicast VLAN.

6.6.3 Configuring IP Multicast VLANs

IP Multicast VLANs use the same interface as other DPF VLANs. DPF VLANs are accessed under the ASRT or SuperELAN Services configuration and console menus. From the configuration menus, a DPF VLAN may be added, deleted, enabled, disabled, changed or listed under the *Talk 6* menus. Similar options are available under *Talk 5* as well as additional options to reset the VLAN counters, save the run-time configuration into SRAM, or load and use the configuration from SRAM.

There is full dynamic support when configuring VLANs as long as the bridge into which DPF is being applied is already defined and running.

The DPF function for IP Multicast VLANs is automatically enabled by default, as is the auto-creation of IP Multicast VLANs. Default settings should work in a

normal environment so there is no need to configure either of them. Nevertheless, IP Multicast VLANs can be manually configured in the same way as other types of VLAN.

To create an IP Multicast VLAN, use the command ADD. The syntax is very similar to that for other types of VLAN. Figure 59 on page 131 shows the configuration sequence required to define an IP Multicast VLAN. The parameters that have to be entered are:

- **IP Multicast Address:** Specifies the IP group address whose multicast traffic will be filtered to create this VLAN. Valid values range from 224.0.1.0 to 239.255.255.255.
- **Configure This VLAN on Specific Ports?:** Answering **yes** leads to a submenu which allows inclusion or exclusion of specific ports for this VLAN. It is equivalent to entering the **Ports** submenu in the config tool.
- **Age (expiration in minutes):** Specifies the amount of time that an Auto-Detect port will remain in the forwarding state in the absence of traffic received from that port for this VLAN. A value of 0 means “Never Expires”. The default value is 10 minutes.
- **Mac Addresses Tracking:** Setting this parameter to **enabled** causes source MAC addresses from transmissions on the VLAN to be saved. These learned addresses can be displayed with the show-members command. Learned addresses will be aged out with the aging timer for this VLAN. The default value is **disabled**.
- **Enable this filter?:** Enables or disables this VLAN filter.
- **VLAN Name:** Specifies the name for this VLAN. Names are used later to reference a VLAN.

```
MSS1 ASRT config>VLANS

VLAN filter configuration for ASRT bridge
MSS1 ASRT VLAN config>ADD IP-MULTICAST
IP Multicast Address [0.0.0.0]? 231.2.3.4
Configure This VLAN on Specific Ports? [No]:
Age (expiration in minutes,0=infinity) [10]?
Track Active Mac Addresses on this VLAN? [No]:
Enable This Filter? [Yes]:
VLAN Name (32 chars max) []? IP Multicast 1
VLAN 'IP Multicast 1' (IP Multicast 231.2.3.4) successfully added
MSS1 ASRT VLAN config>
```

Figure 59. IP Multicast VLANs Configuration

To view current configuration and run-time status within *Talk 5*, use the LIST command. An example of this command is shown in Figure 60 on page 132. To view MAC addresses and ports associated to a VLAN, use the SHOW-MEMBERS IP-MULTICAST command.

```
MSS1 ASRT>VLANS

VLAN filter console for ASRT bridge
MSS1 ASRT VLAN console>LIST IP-MULTICAST ALL

   ----------------------- IP MULTICAST VLANS ------------------------

   IP Multicast Address        = 224.0.0.1
   Port  1 (Interface  1)      = Auto-Detect and Include, Forwarding
   Port  2 (Interface  2)      = Auto-Detect and Include, Not Forwarding
   Port  3 (Interface  3)      = Auto-Detect and Include, Not Forwarding
   Port  4 (Interface  4)      = Auto-Detect and Include, Not Forwarding
   Age (expiration in minutes) = 10
   Tracking of Mac Addresses   = Disabled
   VLAN Status                 = Enabled
   Packets Processed           = 5
   Discards Due To Exclusion   = 0
   VLAN Name                   = Config for Auto-Created VLANs
   ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
   IP Multicast Address        = 231.2.3.4
   Port  1 (Interface  1)      = Auto-Detect and Include, Not Forwarding
   Port  2 (Interface  2)      = Auto-Detect and Include, Not Forwarding
   Port  3 (Interface  3)      = Auto-Detect and Include, Not Forwarding
   Port  4 (Interface  4)      = Auto-Detect and Include, Not Forwarding
   Age (expiration in minutes) = 10
   Tracking of Mac Addresses   = Disabled
   VLAN Status                 = Enabled
   Packets Processed           = 0
   Discards Due To Exclusion   = 0
   VLAN Name                   = IP Multicast 1
MSS1 ASRT VLAN console>
```

Figure 60. Viewing IP Multicast Configuration

6.7 Comparison Between BCM, BBCM, DPF VLANs and BUS Filters

Several methods aimed at reducing unnecessary traffic in the ATM network have been mentioned in this chapter.

BUS data frame filtering is used at the LANE BUS level to allow specific types of data frames to be forwarded or discarded.

Since backbone campus networks have historically carried multiprotocol data traffic, LAN Emulation has provided the most viable ATM migration path from token-ring and Ethernet Legacy networks. LANE relies on the emulation of traditional LANs via the *LAN Emulation Server* (LES) and *Broadcast and Unknown Server* (BUS) to emulate the shared media characteristics of both token-ring and Ethernet. Since ATM does not provide a shared medium, the BUS provides the distribution function for all broadcast, multicast and unknown unicast frames to other devices in the ELAN. Because all broadcast, multicast and unknown unicast frames must flow through the BUS, the BUS is an ideal location for controlling and managing ATM backbone data traffic.

Previously, ATM LANE data filtering had to be supported and configured in an ATM edge device. These devices provide the connectivity between ATM LANE and Ethernet and token-ring. The most common ATM LANE edge device is a bridge, but some devices also support OSI Layer 3 and Layer 4 routing. Because

of the large number of ATM edge devices available, it is not unusual for an ATM campus backbone to have ATM edge devices from multiple vendors. Since each vendor's ATM edge device implementation is different, managing data traffic flows at the edge of the ATM network can be difficult. By providing BUS filters, configuration and management of backbone network filters are centralised in a single MSS Server. Network utilisation can be improved by preventing unwanted data traffic from entering the ATM backbone at all, even from ATM edge devices that do not support filters. Network security can be improved by restricting the type of data traffic allowed on the ATM backbone and which devices the data traffic is destined for and or originated from.

VLANs have been generating considerable interest among users, with the expectation of several potential benefits:

VLANs can improve performance by:

- Reducing broadcast traffic
- Restricting frames to segments running the relevant protocol

VLANs can simplify network management by:

- Removing the need for manual reconfiguration when endstations are moved
- Conserving IP addresses by removing the need for reassignment when endstations are moved

VLANs can improve network security by limiting access to traffic.

A VLAN is a logical subset of a physical network or networks. An ELAN is also a VLAN; it's one of the benefits of ELANs. A VLAN could be based on physical port location, MAC address, network address, or some other defining characteristic. A PVLAN is a VLAN where membership is based on the protocol type.

MSS establishes PVLAN membership dynamically using a technique called Dynamic Protocol Filtering (DPF). DPF monitors the traffic on bridge ports, and forwards traffic for a given subnet only to ports that are actively participating in that subnet. Support is provided for IP, IPX, and NetBIOS. Use of DPF means that no pre-configuration is required in the endstations, and endstations can be moved without reconfiguration.

MSS includes a Broadcast Manager that is a key improvement over standard BUS operation, allowing broadcast frames to be converted to unicast frames whenever possible. Rather than sending all broadcast frames to all users, BCM identifies the specific type of broadcast frame and sends this only to those devices that will be interested in receiving it. This service is provided for IP, IPX, and NetBIOS protocols. The BCM software dynamically identifies frames that can be converted to unicast - special configuration is not required. One customer enjoyed an 80% reduction in IPX broadcast traffic using BCM! Clearly, the less multicast traffic that has to traverse the backbone, the more bandwidth there is available for all traffic. MSS also offers *Intelligent BUS Mode* (IBUS); in IBUS mode unicast frames are sent directly to non-proxy LECs without disturbing other non-proxy LECs.

The Broadcast Manager function introduced in MSS Release 1.0 reduces the number of broadcast frames sent through the BUS. Bridging Broadcast Manager (BBCM), available since MSS Release 1.1, provides a similar function for frames

bridged by MSS. BBCM is implemented for IP and NetBIOS, and converts broadcast frames destined for a learned protocol address to unicast frames.

While IBUS and BCM provide functions which offer attractive benefits, these capabilities do increase the processing required for each packet.

So how do these four methods of reducing unwanted traffic compare?

- The BBCM works between ELANs where broadcasts over the MSS bridge function are changed into unicast traffic.
- The same applies to the BCM function, but now the broadcast traffic is within an ELAN.
- With DPF VLANs the users are placed in the same protocol VLAN and thus the need of broadcasts is getting lower.
- Finally, BUS data frame filtering actually checks if data needs to be forwarded at all or simply discarded altogether, based on defined filters.

Chapter 7. Usability Enhancements

This short chapter highlights the new functions in the latest release of MSS which ease the configuration and manageability of the MSS:

- Command Completion
- Packet Trace Decoding Aids
- Dynamic 1483 PVC/SVC
- Dynamic Reconfiguration
- CPU Performance Monitor
- Non-Zero VPI Support

7.1 Command Completion and General Usability Enhancements

The command-line interface has been re-worked with “usability” in mind, and the following list of changes has been made:

1. Command Completion, in which the full command no longer needs to be typed in. This implementation is very similar to the existing 8260 Control Point Switch. In cases of incomplete commands where various possibilities exist to complete the command, the user will be shown a list of possible commands and can cycle through the available commands using the TAB key as an alternative to typing any of the alternatives explicitly; Figure 61 on page 147 shows an example of this in use.
2. Online help to point the novice user in the right direction.
3. OPCON menu has been rearranged, with the less commonly-used commands placed at the bottom of the list.
4. A secondary ELS Console can be found using *Talk 7*. This is especially useful if the user is in the middle of a configuration or monitoring process using *Talk 5* or *Talk 6*; it is no longer necessary to exit this process to invoke ELS but instead Ctrl-P followed by *Talk 7* can be used. Once ELS changes have been made, the user can return to the point where he was using *Talk 5* or *Talk 6*.
5. The “retrieve” shortcut command Ctrl-B works in the OPCON menu itself.
6. Ctrl-U can be used to “undo” commands; this is actually not a new command but is now documented clearly.
7. If the user enters `EXIT` when at the top level of the *Talk 5* or *Talk 6* menu structure he will be prompted to use Ctrl-P to return to OPCON.
8. Ctrl-W backs up by one word.
9. --More-- is added for long displays; pressing the Enter key advances by one line whereas the spacebar advances by one page.
10. Additional help - typing “p?” will list commands which start with the letter p.
11. Command Completion can be disabled although it is enabled by default.
12. The existing Ctrl-B “retrieve” command will now only retrieve commands which are consistent in the context in which the user is currently working.
13. Some synonyms are known and cause a prompt for a possible alternative command when the original command is invalid in the particular context: for example, LIST, SHOW, DISPLAY.
14. Entering the command ’;’ allows the next line to be entered as-is, for example as a comment for a log; in other words, Command Completion is disabled for one command.

Note that some other enhancements have been considered and rejected: for example, it would have been possible to define the EXIT command at the top level of *Talk 5* or *Talk 6* as the method of returning to the OPCON prompt, but it was decided that too many automated operators would have been broken by this change. For example, some of these enter a long succession of EXIT commands to ensure they are in fact at the top level of the *Talk 5* or *Talk 6* menu structure.

7.2 Packet Trace Decoding Aids

Packet Trace for LAN Emulation Clients, the LAN Emulation Server and for Classical IP has been enhanced to decode Layer 3 and some Layer 3/4 information. Access this function using the command sequence:

```
TALK 5
EVENT
PACKET
```

The previous release only provided a hexadecimal dump which had to be decoded manually offline.

Use the command SET DECODE to enable; for example:

```
SET DECODE INCLUDE IP
SET DECODE EXCLUDE IPX
```

Additional search tools have been provided to improve on-line viewing of the decoded trace, using the commands VIEW, SEARCH and TRACE-STATUS.

7.3 Dynamic 1483 PVC/SVC

This function was originally provided as a special code build for the 2216 and allows activation of a PVC or SVC for RFC 1483. It has now been incorporated in the base MSS code. The purpose of this feature is to allow activation of configured Classical IP and IPX over RFC 1483 permanent PVC/SVC entries without having to reboot the box or to reset the interface. Prior to this design, a permanent PVC/SVC is configured in *Talk 6*, then the box needs to be rebooted or the interface is reset to activate the configured entry(s). It is no longer necessary to reboot or reset the interface to enable this new PVC/SVC. The newly configured permanent entries can be activated from *Talk 5*. Access this function with a command sequence such as:

```
TALK 6
PROTOCOL ARP
ADD PVC

ACT CHAN SPEC 0
```

7.4 Dynamic Reconfiguration

Each release of “Common Code for IBM’s router platforms increases the number of parameters that can be modified without the need to reload the router completely, and MSS benefits from this work.

7.4.1 General Description

7.4.1.1 Summary

Dynamic Reconfiguration implements the following functions:

- *Activate interface* - provides the capability to configure and activate a new interface, including the protocols and features configured to run on the interface, without restarting the router. This function affects both the configuration memory and the current run-time environment.
- *Reset interface* - provides the capability to disable an existing interface in the current run-time environment, including the protocols and features running on the interface, and then automatically reactivate the interface, including the protocols and features running on the interface, using new interface-based configuration parameters. This function affects both the configuration memory and the current run-time environment.

  Note: Not all components are providing this capability. See 7.4.2.2, “Reset Interface” on page 141 for more information.
- *Reset protocol* - provides the capability to disable a protocol on an existing interface in the current run-time environment (or globally) and then automatically reactivate the protocol on the interface (or globally) using new interface-based (and global) configuration parameters. This function affects both the configuration memory and the current run-time environment.

  Note: Not all components are providing this capability. See 7.4.2.2, “Reset Interface” on page 141 for more information.
- *Reset parameter* - provides the capability to reset protocol or feature parameters either individually or on a sub-component basis (where interdependencies exist between parameters). A common set of commands is used to ensure consistency between nets, protocols and features. This function affects both the configuration memory and the current run-time environment.

  Note: Not all components are providing this capability. See 7.4.2.2, “Reset Interface” on page 141 for more information.

7.4.1.2 Limitations

The following *activate interface* restrictions still exist:

- Dynamically removing or replacing a device is not supported.
- Activating a protocol on a new interface for the first time is not supported (the protocol must be previously enabled on some other interface before the *activate interface*).
- Activating a new interface which has a header or trailer larger than the currently allocated global buffers requires a restart.
- *Delete interface* must be followed by a router restart before an *activate interface* is issued.

The following are *reset interface* restrictions:

- Dynamically changing the device hardware type is not supported.
- Dynamically changing the DLC type is not supported.

- The MTU size of the reset interface cannot be larger than before the reset because private receive buffers are not reallocated. Similar restrictions exist for header and trailer sizes.
- Enabling a protocol on an interface for the first time is not supported.
- *Delete interface* must be followed by a router restart before an *reset interface* is issued.

See 7.4.2.1 below and 7.4.2.2, “Reset Interface” on page 141 for further restrictions and limitations.

7.4.2 Detailed Description

7.4.2.1 Activate Interface

Activate interface provides the capability to configure and activate a new interface, including the protocols and features configured to run on the interface, without restarting the router. This can be useful when defining new virtual interfaces, such as ATM LAN emulation clients.

The new interface is configured, along with the protocols and features, using the `CONFIG` process (*Talk 6*) commands. These commands affect the contents of the configuration memory. The configuration changes are activated by issuing the `GWCON` process (*Talk 5*) *activate interface* command.

The run-time parameters of the other interfaces and the protocols and features running on them are not affected by the *activate interface*.

When a new interface is activated, it replaces one of the spare interfaces in the current run-time environment. By default, there are no spare interfaces in the current run-time environment. To configure the number of spare interfaces:

- Access the **CONFIG** (*Talk 6*) process
- Use the `set spare-interfaces` command to set the number of spare interfaces
- Exit the **CONFIG** process by pressing *Ctrl-P*
- Restart the router

```
Example:  *talk 6
            Config>set spare 2
            Config>
            *restart
            Are you sure you want to restart the gateway? (Yes or [No]): yes
```

To display the configured number of spare interfaces:

- Access the **CONFIG** (*Talk 6*) process
- Use the `list configuration` command to list the configuration, including the number of spare interfaces

```
Example:  *talk 6
            Config>list config

            . . . configuration list . . .

            Number of spare interfaces: 2

            . . . configuration list . . .
```

```
Config>
```

The default number of spare interfaces is 0. The maximum number of spare interfaces is 99.

Spare interfaces are installed as NULL devices during router initialisation. When referring to interfaces by number, spare interfaces follow the real and virtual interfaces and precede the bridge net and IP tunnel interfaces. Each time the router restarts, a new set of spares are installed as NULL devices.

To display the spare interfaces,

- Access the **GWCON** (*Talk 5*) process
- Use the `configuration` command to list the run-time configuration, including the spare interfaces

```
Example:  *talk 5
            +config

            . . . configuration list . . .

            Net Interface  MAC/Data-Link         Hardware        State
            0   TKR/0      Token-Ring/802.5      ATM             Up
            1   TKR/1      Token-Ring/802.5      ATM             Disabled
            2   NULL/0     Null device           None            Not present
            3   NULL/1     Null device           None            Not present

            +
```

To activate a new interface (that is, to use one of the spares):

- Access the **CONFIG** process (*Talk 6*)
- Use the `add device` command to add an interface
- Use the `net` command and other commands to configure the interface or to add ATM LAN emulation clients
- Use the `protocol` and `feature` commands to configure the various protocols and features on the interface
- Use the `write` command to write the configuration to the configuration memory if this is required by the router
- Exit the **CONFIG** process by pressing *Ctrl-P*
- Access the **GWCON** process (*Talk 5*)
- Use the `activate` command to activate the new interface and the protocols and features on the interface

The following tables use the legend below for Support:

- actifc - activate interface
- MIN - minimum required support is provided (that is, indication that a restart is required)
- None - no support (not even MIN) has been committed

- N/A - no support required

Table 9. Device Driver Support for Dynamic Activation

Component	Support	Restrictions
8210 155M ATM	N/A	None
8210 FDDI	N/A	None

Table 10. Net Handler Support for Dynamic Activation

Component	Support	Restrictions
FasTR token-ring	actifc	None
FDDI	actifc	None
ATM Net	MIN	Cannot activate ATM interface
ATM Virtual Interface	actifc	ATM base net must already be active
ATM LEC/PLEC	actifc	ATM base net must already be active
ATM Virtual Network (NHRP) Interface	N/A	None

Table 11. Protocol Support for Dynamic Activation

Component	Support	Restrictions
IP, ICMP, UDP, IP Host	actifc	None
RIP	actifc	None
OSPF, MOSPF	actifc	None
BGP	N/A	Use the BGP reset neighbor command to activate new neighbours
DVMRP	actifc	None
TCP	N/A	None
Telnet, TFTP	N/A	None
Bridge	actifc	Must restart if Bridge personality (SR, SR-TB, SRT, TB) is changed by newly activated interface Must restart if any of the following bridge parameters have changed: bridge enable, FA-GA conversion enable, virtual segment enable, LF bit interpretation, SRTB conversion enable, 8209 spanning tree enable, DLS enable, UB encapsulation enable
Bridge Filters, NetBIOS Filters	actifc	Must restart if NetBIOS filters configured on interface
MAC Filters	actifc	MAC filters on all interfaces re-initialised
LNM	MIN	Must restart if LNM configured on interface
ARP	actifc	None

Component	Support	Restrictions
IPX	actifc	Use the IPX reset filter-lists command to reset interface-based filters configured on the interface. Use the IPX reset route-static and IPX reset sap-static commands to reset IPX static routes and static services configured on the interface
AppleTalk	actifc	None
Banyan Vines	actifc	None
DLSw, BAN, NetBIOS	actifc	None
APPN	actifc	Activate interface first, then configure APPN on interface
LLC	actifc	None
SNMP	actifc	None
ATM LES/BUS/BCM	actifc	None
ATM ARP	actifc	None
ATM RFC1483	actifc	None
ATM SCSP	actifc	None
ATM MARS	MIN	Must restart
ATM NHRP	N/A	None

7.4.2.2 Reset Interface

reset interface provides the capability to disable an existing interface in the current run-time environment, including the protocols and features running on the interface, and then automatically reactivate the interface, including the protocols and features running on the interface, using new interface-based configuration parameters for the interface and protocols and features running on the interface. This function affects both the configuration memory and the current run-time environment.

The interface, protocols and features configuration parameters are changed using the CONFIG process (*Talk 6*) commands. These commands affect the contents of the configuration memory. The configuration changes are activated by issuing the GWCON process (*Talk 5*) reset interface command.

The run-time parameters of the other interfaces and the protocols and features running on them are not affected by the *reset interface*.

To reset an existing interface:

- Access the **CONFIG** process (*Talk 6*)
- Use the `net` command and other commands to change configuration parameters
- Use the `protocol` and `feature` commands to change the interface-based configuration parameters

- Use the `write` command to write the configuration to the configuration memory if this is required by the router
- Exit the **CONFIG** process by pressing *Ctrl-P*
- Access the **GWCON** process (*Talk 5*)
- Use the `reset interface` command to reset the interface, the protocols and features on the interface

7.4.2.3 Reset Protocol

Reset protocol provides the capability to disable an existing protocol or feature on an interface (or globally) in the current run-time environment, and then automatically reactivate the protocol or feature on the interface (or globally) using new interface-based (or global) protocol configuration parameters. This function affects both the configuration memory and the current run-time environment.

The protocol and feature configuration parameters are changed using the CONFIG process (*Talk 6*) commands. These commands affect the contents of the configuration memory. The configuration changes are activated by issuing the GWCON process (*Talk 5*) protocol `reset` command.

When resetting a protocol or feature on an interface, rather than globally, the run-time parameters of the protocol or feature on other interfaces in the run-time environment are not affected. Resetting a protocol or feature should not affect other protocols or features, and should not affect the underlying network interfaces.

To reset an existing protocol or feature on an interface in the current run-time environment:

- Access the **CONFIG** process (*Talk 6*)
- Use the `protocol` or `feature` command to change the interface-based configuration parameters
- Use the `write` command to write the configuration to the configuration memory if this is required by the router
- Exit the **CONFIG** process by pressing *Ctrl-P*
- Access the **GWCON** process (*Talk 5*)
- Use the `protocol` or `feature` command to access the monitoring environment for the protocol or feature
- Use the `reset interface` command to reset the protocol or feature on the interface.

```
Example: *talk 6
          Config>protocol ipx
          IPX Config>

          . . . change interface-based IPX parameters . . .

          IPX>exit
          Config>
          *talk 5
          +protocol ipx
          IPX>reset interface 1
```

To globally reset an existing protocol or feature in the current run-time environment:

- Access the **CONFIG** process (*Talk 6*)
- Use the `protocol` or `feature` command to change the configuration parameters
- Use the `write` command to write the configuration to the configuration memory if this is required by the router
- Exit the **CONFIG** process by pressing *Ctrl-P*
- Access the **GWCON** process (*Talk 5*)
- Use the `protocol` or `feature` command to access the monitoring environment for the protocol or feature
- Use the `reset` command to reset the protocol or feature on the interface.

```
Example: *talk 6
          Config>protocol ospf
          OSPF Config>

          . . . change OSPF parameters . . .

          OSPF>exit
          Config>
          *talk 5
          +protocol ospf
          OSPF>reset ospf
          Reset OSPF subsystem? [Yes]: yes
```

7.4.2.4 Reset Selected Parameters

Rreset parameter provides the capability to reset interface, protocol or feature parameters either individually or on a sub-component basis. Resetting parameters on a sub-component basis is required because some parameters are not strictly interface based or contain interdependencies which cannot be untangled on a *reset interfac*e operation.

Examples of sub-components are protocol filters and static routes.

The configuration parameters are changed using the CONFIG process (*Talk 6*) commands. These commands affect the contents of the configuration memory. The configuration changes are activated by issuing the GWCON process (*Talk 5*) `reset` command under the monitoring console of the affected interface, protocol or feature. Parameters on the `reset` command identify the sub-component being reset.

To reset parameters on a sub-component basis:

- Access the **CONFIG** process (*Talk 6*)
- Use the `net`, `protocol` or `feature` command to change the configuration parameters
- Use the `write` command to write the configuration to the configuration memory if this is required by the router
- Exit the **CONFIG** process by pressing *Ctrl-p*
- Access the **GWCON** process (*Talk 5*)

- Use the `net`, `protocol` or `feature` command to access the monitoring environment for the interface, protocol or feature
- Use the `reset` command to reset the sub-component parameters.

```
Example:  *talk 6
            Config>protocol ipx
            IPX Config>

            . . . enter commands to change IPX SAP filters . . .

            IPX Config>exit
            Config>

            . . . press Ctrl-p to exit CONFIG process . . .

            *talk 5
            +protocol ipx
            IPX>reset filters
```

Configuration parameters changed in the CONFIG process are not automatically activated in the run-time environment. Some overt action on the part of the user must occur in order for the changes to be activated. There are several reasons why:

- Several commands may have to be issued to complete the configuration change. Trying to automatically activate changes on a command-by-command basis may result in missing parameters, parameter conflicts, etc.
- The user may not want the changes to take effect immediately, but rather have them take effect during some scheduled maintenance period.
- It makes it more difficult to incorporate changes made via the configuration tool that have been imported into the router. In this case the new configuration can be imported, and then activated.

Although generally discouraged, parameters changed on a command-by-command basis may take effect immediately in the following cases:

- The command is issued in the CONFIG process and the user is first prompted as to whether the changes should take effect immediately. The default answer should be `no`.
- The command is issued in the GWCON process and does *not* change the configuration memory.

Tables in the sections that follow use the following legend for Support:

- rstifc - reset interface
- prtifc - reset protocol on interface
- prtglb - reset protocol globally
- rstprm - reset parameters
- MIN - minimum required support is provided (that is, an indication that a restart is required)
- None - no support (not even MIN) has been committed

- N/A - no support required

Table 12. Device Driver Support for Dynamic Resetting

Component	Support	Restrictions
8210 155M ATM	MIN	Cannot reset ATM interface
8210 FDDI	rstifc	None

Table 13. Net Handler Support for Dynamic Resetting

Component	Support	Restrictions
FasTR Token Ring	rstifc, prtifc	None
FDDI	rstifc, prtifc	None
ATM Net	MIN, prtifc	Cannot reset ATM interface
ATM Virtual Interface	rstifc, prtifc	None
ATM LEC/PLEC	rstifc, prtifc	None
ATM Virtual Network (NHRP) Interface	MIN	Cannot reset ATM Virtual Network Interface

Table 14. Protocol Support for Dynamic Resetting

Component	Support	Restrictions
IP, ICMP, UDP, IP Host	rstifc, prtifc	In addition to supporting the reset interface command, IP provides its own global reset command which currently resets IP interface parameters, access controls, packet filters. Currently, static routes are dynamic when changed via the console, but not via the config tool.
RIP	rstifc, prtifc	The IP reset command resets all RIP interfaces at once
OSPF, MOSPF	rstifc, prtglb	In addition to supporting the reset interface command, OSPF provides its own reset ospf command
BGP	rstprm	Use the BGP reset neighbor command to reset neighbors
DVMRP	N/A	DVMRP is already dynamic when config changes are made via the console, but not via the config tool
TCP	N/A	None
Telnet, TFTP	N/A	None
Bridge	rstifc	Same as activate interface
Bridge Filters, NetBIOS Filters	rstifc	Same as activate interface
MAC Filters	rstifc, prtglb	Use the reinit command to globally reset all MAC filters
LNM	MIN	Cannot reset LNM
ARP	rstifc, prtifc	None

Component	Support	Restrictions
IPX	rstifc, prtifc, rstprm	In addition to supporting the reset interface command, IPX provides its own reset interface, reset filters, reset filter lists reset access controls, reset route static and reset sap static commands.
AppleTalk	MIN	Cannot reset AppleTalk
Banyan Vines	MIN	Cannot reset Banyan Vines
DLSw, BAN, NetBIOS	rstifc	None
APPN	rstifc, prtglb	For talk 5, reset interface; APPN only picks up changes to a couple of network interface parameters. Use the APPN activate new config command to reset APPN parameters.
LLC	rstifc	None
SNMP	MIN	Use the SNMP revert command to reset global SNMP parameters
ATM LES/BUS/BCM	rstifc	None
ATM ARP	rstifc, prtifc	None
ATM RFC1483	rstifc, prtifc	None
ATM SCSP	rstifc, prtifc	None
ATM MARS	MIN	Cannot reset MARS
ATM NHRP	MIN	Must restart

7.5 CPU Performance Monitor

MSS again benefits from enhancements to Common Code which can best be shown by demonstrating its configuration and use in the following figures:

```
MSS1 *TALK 6
Gateway user configuration
MSS1 Config>PERFORMANCE
Performance Monitoring Configuration
MSS1 PERF Config>?

Possible completions:
 ...    DISABLE
 ...    ENABLE
        LIST
 ...    SET
        EXIT
(you may cycle through these commands by pressing the TAB key)
MSS1 PERF Config>
MSS1 PERF Config>LIST
CPU Monitor configuration:
CPU Monitor State:                DISABLED
Short Window (SW) time (secs)     5
Packet Statistics:                DISABLED
Link Utilization Statistics       DISABLED
T2 window output:                 DISABLED
MSS1 PERF Config>ENABLE

Command not fully specified
Possible completions:
        CPU statistics
        LINK statistics
        PACKET statistics
        T2 output
(you may cycle through these commands by pressing the TAB key)
(hit Carriage Return again to abort command)
MSS1 PERF Config>ENABLE CPU
Enable/Disable state successfully set.
MSS1 PERF Config>ENABLE LINK
Link Utilization statistics Enable/Disable state successfully set.
The CPU monitor must be enabled before packet statistics will be shown!
MSS1 PERF Config>
```

Figure 61. Configuring the Performance Monitor Using Talk 6

Figure 61 shows how to configure the Performance Monitor using *Talk 6*. The configuration parameters shown here will not be enabled until the MSS is reloaded.

```
MSS1 *TALK 5

CGW Operator Console

MSS1 +PERFORMANCE
Performance Monitoring Console
MSS1 PERF Console>?

Possible completions:
        CLEAR statistics
 ...    DISABLE
 ...    ENABLE
        LIST
        REPORT
 ...    SET
        EXIT
(you may cycle through these commands by pressing the TAB key)
MSS1 PERF Console>
MSS1 PERF Console>REPORT
Statistics not available yet, please try again later
MSS1 PERF Console>ENABLE

Command not fully specified
Possible completions:
        CPU statistics
        LINK statistics
        PACKET statistics
        T2 output
(you may cycle through these commands by pressing the TAB key)
(hit Carriage Return again to abort command)
MSS1 PERF Console>ENABLE CPU
CPU monitor enabled.
MSS1 PERF Console>REPORT
Statistics not available yet, please try again later
MSS1 PERF Console>ENABLE PACKET

Packet statistics Enabled.
(The CPU statistics are a prerequisite and must also be enabled!)
MSS1 PERF Console>ENABLE LINK

Link utilization and packet statistics have been enabled.
Link utilization statistics are viewable ONLY with the REPORT command.
The CPU statistics are a prerequisite and must also be enabled!
MSS1 PERF Console>LIST
CPU Monitor State:               ENABLED
Short Window (SW) time (secs)    5
Packet Statistics:               ENABLED
Link Utilization Statistics      ENABLED
T2 window output:                DISABLED
```

Figure 62. Configuring the Performance Monitor Using Talk 5

Figure 62 shows the equivalent commands to the ones used in Figure 61 on page 147, the difference being that the Performance Monitor settings made using *Talk 5* take effect immediately.

```
MSS1 *TALK 5

MSS1 PERF Console>REPORT
--------------------------------------------------------------------------------
KEY:  SW = Short Window =  5 seconds
KEY:  LW = Long Window =   5.0 minutes  (60 x SW)

  CPU LOADING : Most recent SW                    =   0%
                Most recent LW                    =   0%
                Highest for all SW's              =   0%
                Highest for all LW's              =   0%
                % of time loading (SW) was > 60%  =   0%
                % of time loading (SW) was > 70%  =   0%
                % of time loading (SW) was > 80%  =   0%
                % of time loading (SW) was > 90%  =   0%
                % of time loading (SW) was > 95%  =   0%
--------------------------------------------------------------------------------
               RECENT     HIGH       RECENT     HIGH        ACCUM     ACCUM
  INTERFACES   RX UTIL    RX UTIL    TX UTIL    TX UTIL     RX LOSS   TX LOSS
  SLOT  PORT   SW   LW    SW   LW    SW   LW    SW   LW     (pkts)    (pkts)
--------------------------------------------------------------------------------
   01    01     0%   0%    0%   0%    0%   0%    0%   0%         0         0
--------------------------------------------------------------------------------
MSS1 PERF Console>
MSS1 *logout
```

Figure 63. Sample Performance Report

Figure 63 shows how little the MSS running in our test environment is actually being used!

7.6 Non-Zero VPI Support

The change to allow specification of VPI numbers in the range 0-7, which may be required when connecting to public ATM service providers, has the side-effect that the maximum VCI number supported in the MSS has changed to 16K, from its previous maximum value of 32K. It's not felt that this will cause any significant impact.

7.7 Conclusion

Our experience is that the enhancements that are described in this section, in particular those under the heading "usability", do indeed make the MSS Server significantly easier to use.

Chapter 8. MSS Release 2.2 LANE Redundancy Enhancements

MSS Release 2.2 provides a number of enhancements to existing *redundancy* features. This chapter lists these features, explains how to configure them and gives some examples of how and where they would be useful to implement in a user network.

8.1 Summary

The following is a list of all the new features and enhancements in MSS Release 2.2 which impact redundancy within the MSS:

- LES/BUS Enhanced Redundancy
- LES/BUS Peer Redundancy
- LECS Database Synchronization
- Persistent Data Direct VCCs

For a list of redundancy features which were available in prior releases, see 4.2, “MSS Family History (Software)” on page 74.

8.2 LES/BUS Enhanced Redundancy

MSS Release 2.2 includes two new features to improve the LES/BUS redundancy capability of the MSS, *LES/BUS Enhanced Redundancy* and *LES/BUS Peer Redundancy*. LES/BUS Enhanced Redundancy reduces the time taken to detect the failure of the primary LES/BUS, and avoids situations where a failed primary LES/BUS goes unnoticed. LES/BUS Peer Redundancy reduces the network outage caused by a failure of the primary LES/BUS. This section discusses Enhanced Redundancy; Peer Redundancy is covered in the following section.

8.2.1 Previous Implementation

LES/BUS Redundancy has been available since MSS Release 1.0. This relies on the existence of a *Redundancy VCC* between primary and backup LES/BUS to imply that the other LES/BUS was operational. While the Redundancy VCC remains active the backup LES/BUS assumes that the defined primary LES/BUS is active and will therefore not accept LEC registration itself.

In the event of a primary LES/BUS failure, the Redundancy VCC will be dropped, at which point the backup LES/BUS will accept LEC registration. The LAN Emulation Clients that were previously registered with the primary LES/BUS will detect its failure by released *Control Direct* VCCs and will return to the *LAN Emulation Configuration Server* (LECS). The LECS will then provide the LAN Emulation Clients with the address of the currently active (in this case the backup) LES/BUS. Note that without *Persistent Data Direct VCCs* (see 8.5, “Persistent Data Direct VCCs” on page 167), the architecture compels the affected LAN Emulation Clients to drop all their active sessions when they lose their connections to the LES/BUS.

When the primary LES/BUS returns, the Redundancy VCC will re-establish and the backup LES/BUS will immediately drop all its Control Direct VCCs, causing

the LAN Emulation Clients to return again to the LECS for the address of the currently active (in this case the primary) LES/BUS.

In most cases when the primary LES/BUS fails, the Redundancy VCC will be released quickly, allowing detection of the failure by the backup LES/BUS and quick takeover of the served LAN Emulation Clients. However, there are some failure situations that will result in the VCC remaining up although the LES/BUS is not active, and in that case the Backup LES/BUS will not accept connection requests from the LAN Emulation Clients which have been disconnected from the primary LES/BUS. This problem has been rectified by the new implementation.

8.2.2 New Implementation

Enhanced Redundancy utilizes an *Enhanced Redundancy VCC* between primary and backup LES/BUS. This VCC uses a different SNAP Protocol Identifier (PID) to distinguish it from an old style Redundancy VCC. Enhanced Redundancy Support gets around the problem just mentioned by having each LES/BUS transmit a status message to its partner every two seconds. Overdue status messages can indicate a problem in the partner LES/BUS, even though the Enhanced Redundancy VCC appears to be up. With Enhanced Redundancy, a LES/BUS will take over from its partner if two consecutive status messages are overdue. Thus the time required to detect the failure of a partner LES/BUS is at most six seconds even in obscure failure modes. A pictorial representation of this can be seen in Figure 64.

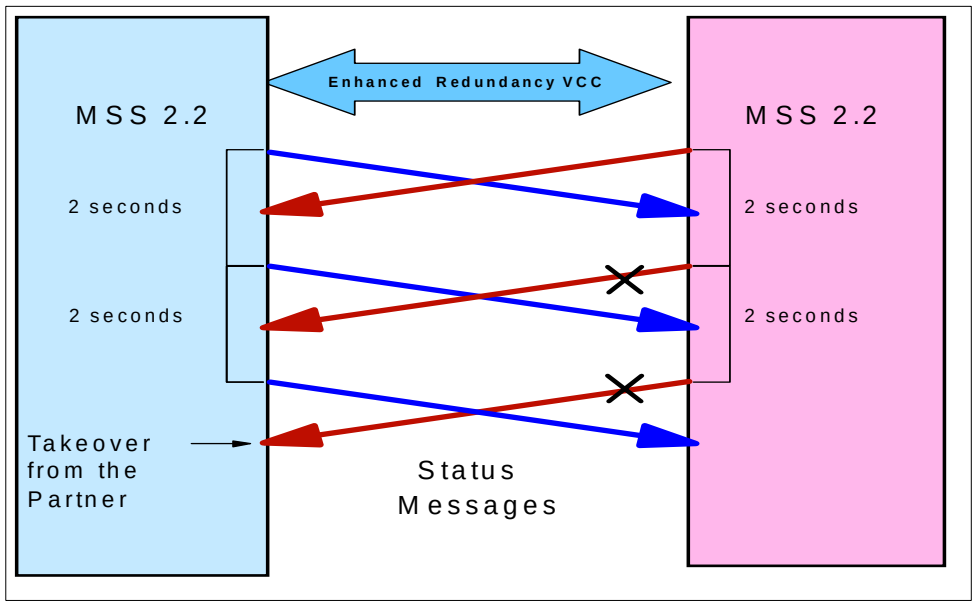


Figure 64. Enhanced Redundancy Status Messages

8.2.3 LES/BUS Enhanced Redundancy Configuration

There are no additional configuration requirements from the previous version of LES/BUS redundancy. If both the primary and the backup LES/BUS support the new Enhanced Redundancy VCC, the enhanced version will be used; if either LES/BUS is unable to support the Enhanced Redundancy VCC (for example, back-level code), then the old Redundancy VCC will continue to be used.

The steps required to configure LES/BUS redundancy are as follows:

1. Configure the primary and backup LES/BUS each on one of two MSSs, remembering to add the ATM address of the partner LES/BUS.
2. Configure a LECS.
3. Configure all LE Clients to register with the LES/BUS via the LECS (because the LECS is responsible for providing the ATM address of the currently active LES/BUS, a LEC must use the LECS).

8.2.4 LES/BUS Enhanced Redundancy Use

LES/BUS redundancy can be used in any situation in which two or more MSSs are installed in a network and it is desired to increase availability of the network by allowing one MSS to provide a backup for functions running in the other. LES/BUS Enhanced Redundancy will be used automatically when Redundant LES/BUS is configured (see above).

8.2.5 LES/BUS Enhanced Redundancy Migration Considerations

There are no migration considerations for this enhancement. The enhanced version is fully compatible with previous versions. Figure 65 shows the Redundancy VCC setup, first between two MSS Release 2.2 LES/BUSes and then between LES/BUSes running dissimilar MSS releases.

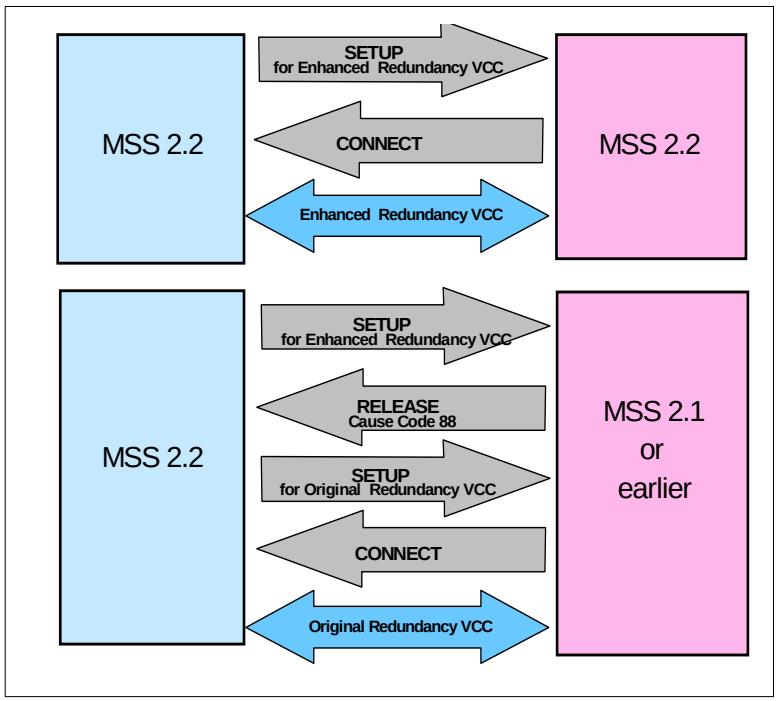


Figure 65. Redundancy VCC Establishment

8.3 LES/BUS Peer Redundancy

LES/BUS Peer Redundancy is a further enhancement to the previous LES/BUS redundancy feature. The purpose is to reduce the network outage caused by a failure of the primary LES/BUS.

8.3.1 Previous Implementation

From the description of LES/BUS redundancy in the previous section it can be deduced that a single failure typically results in two disruptions to the Emulated LAN: one when the primary LES/BUS becomes unavailable, and another when the primary LES/BUS becomes available again and the backup LES/BUS yields to the primary.

8.3.2 New Implementation

LES/BUS Peer Redundancy avoids the second disruption in most cases by allowing the backup LES/BUS to retain its LE Clients even after the primary LES/BUS becomes available. When the primary recovers, after a failure, the primary and backup exchange information every two seconds to determine which LES/BUS will serve the ELAN.

When the Enhanced Redundancy VCC is established:

- If the backup LES/BUS has been operational for less than one minute, the backup automatically yields to the primary. This ensures that a restart of both LES/BUSes at the same time will result in the LES/BUS defined as “primary” taking control in the first place.

Note

Don’t get confused, as we did, with the wording “operational” above. It does *not* mean “If the backup LES/BUS has been serving as active LES/BUS by taking over from the primary LES/BUS for less than a minute” but means, rather, exactly what it says - the backup LES/BUS will only yield its clients to the primary under these circumstances if the Redundancy VCC is established within a minute of the backup LES/BUS starting up. This condition only applies when both primary and backup LES/BUS are started at the same time.

- The primary LES/BUS will always take over if the primary and backup have an equal number of operational LE Clients.
- If an active backup LES/BUS loses all of its clients, the primary LES/BUS will take over since both LES/BUSes would have zero clients.

Figure 66 on page 155 shows an example of Peer Redundancy in operation in the case of primary LES/BUS failure, assuming that the network has been in operation for more than one minute prior to the failure and recovery of the primary LES/BUS.

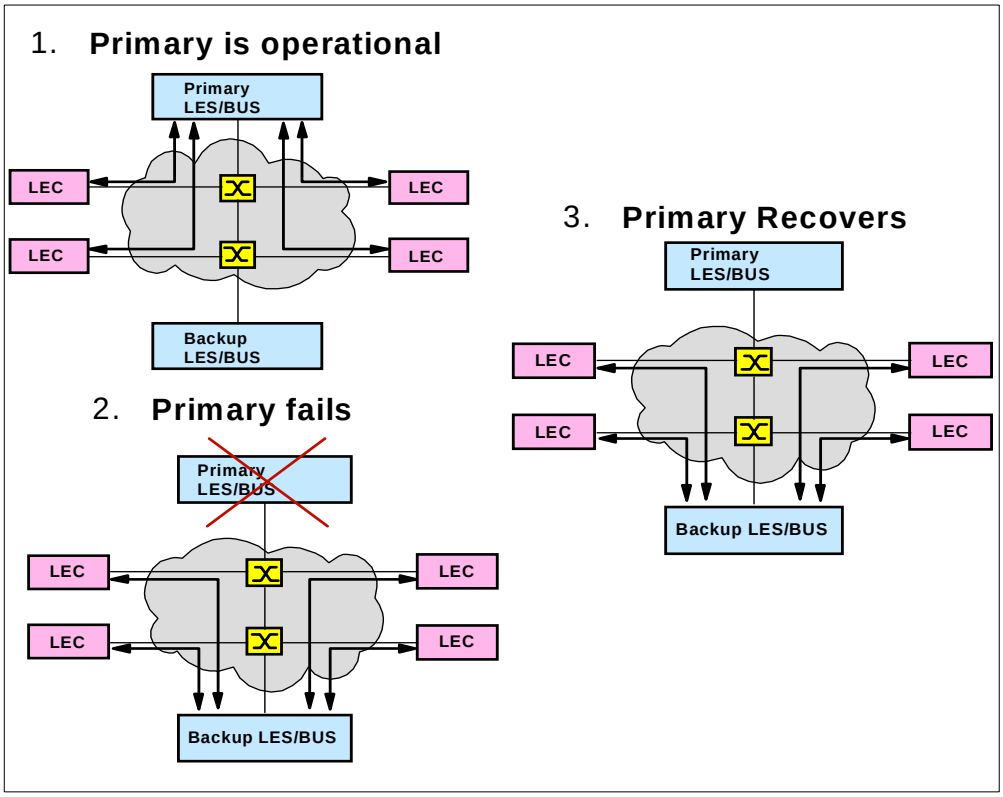


Figure 66. Primary LES/BUS Failure

Figure 67 shows an example of Peer Redundancy in operation in the case of ATM link failure.

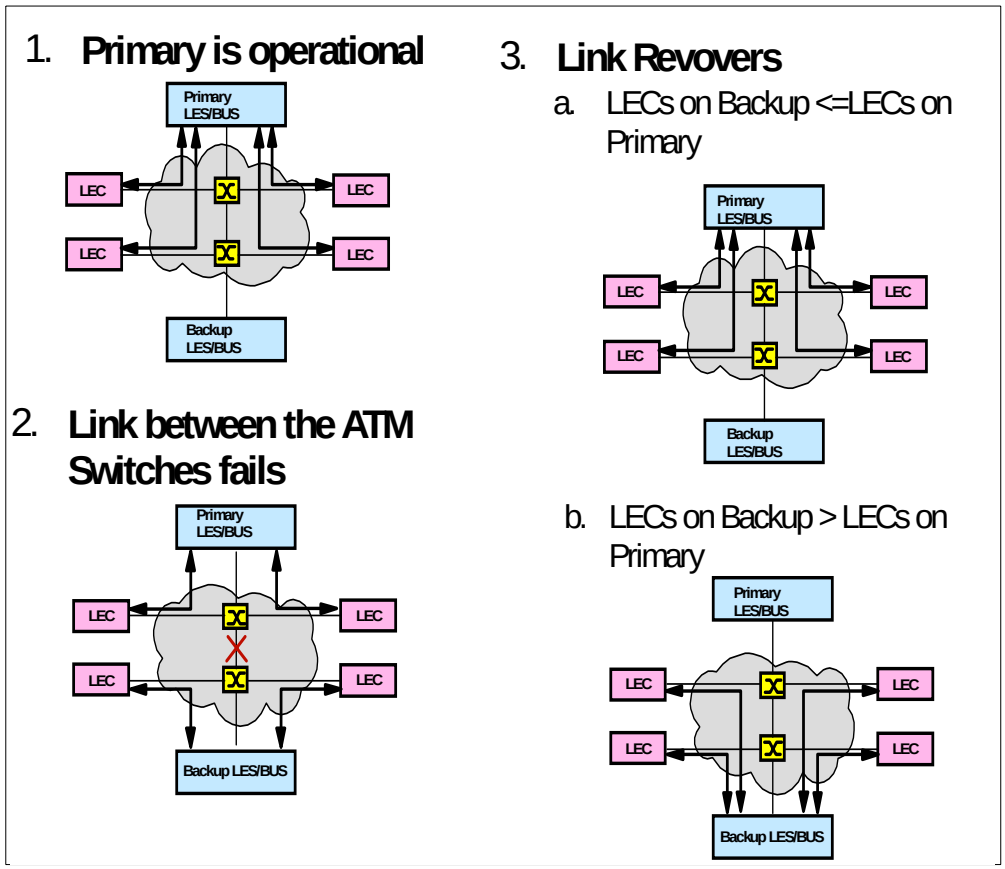


Figure 67. ATM Link Failure

8.3.3 LES/BUS Peer Redundancy Configuration

A new LES/BUS configuration option allows Peer Redundancy to be enabled in addition to the original LES/BUS redundancy parameters. A LES/BUS with redundancy enabled is configured as either a “primary” or “backup”. For simplicity, only the primary LES/BUS has the new Peer Redundancy option. As before, the primary is configured with the ATM address of the backup LES/BUS. Peer Redundancy is disabled by default to maintain the same LES/BUS redundancy behaviour as in prior releases of the MSS Server.

8.3.3.1 Configuration Steps:

1. Configure the primary LES/BUS, including the new Peer Redundancy option (see below).
2. Configure the backup LES/BUS, remembering to add the ATM address of the partner LES/BUS.
3. Configure a LECS.
4. Configure all LE Clients to register with the LES/BUS via the LECS, because the LECS is responsible for providing the ATM address of the currently active LES/BUS.

8.3.3.2 Configuring Using the Configuration Tool

Figure 68 on page 157 shows the navigation path to the screen from where Peer Redundancy is selected. Figure 69 on page 158 shows the configuration window.

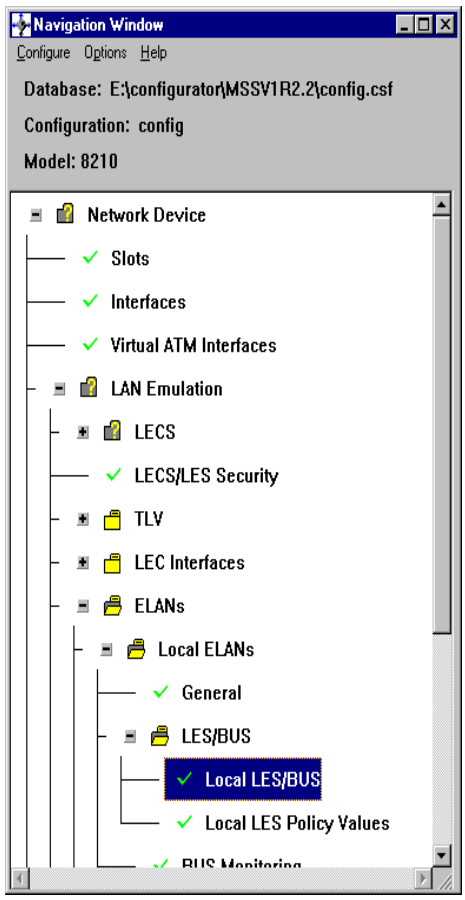


Figure 68. Navigation Path for Peer Redundancy

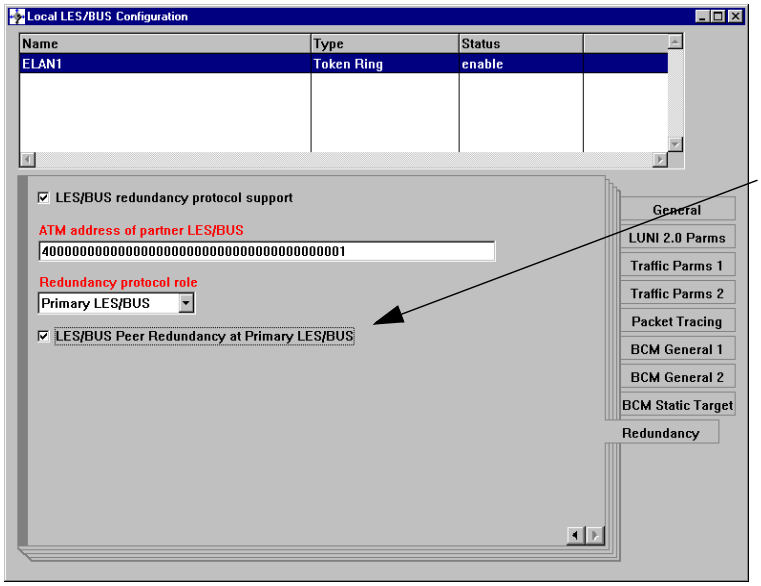


Figure 69. Enabling Peer Redundancy through the Configuration Tool

8.3.3.3 Configuring Using Talk 6

Figure 70 shows the Peer Redundancy option using command line configuration.

```
MSS1 *TALK 6
Gateway user configuration
MSS1 Config>NETWORK 0
ATM user configuration
MSS1 ATM Config>LE-SERVICES
LAN Emulation Services user configuration
MSS1 LE Services config>LES-BUS 4
LES-BUS configuration
MSS1 LES-BUS config for ELAN 'TKR_1'>ENABLE REDUNDANCY
  (  1) Primary LES-BUS
  (  2) Backup LES-BUS
Redundancy Role [0]? 1
Enable Peer Redundancy Option? [No]: y      1
ATM address of backup LES-BUS []?
40:00:00:00:00:00:00:00:00:00:00:00:00:00:0
0:00:00:00:00
MSS1 LES-BUS config for ELAN 'TKR_1'>
```

Figure 70. Peer Redundancy Option

1 This is the new option, Y = Enable

8.3.3.4 Configuring Using Talk 5

The *Talk 6* command above is also available under *Talk 5* for dynamic configuration.

8.3.3.5 Confirming Correct Configuration/Operation

Using *Talk 5* on the MSS with the primary LES/BUS, Peer Redundancy can be seen to be configured using the following command sequence:

```
talk 5
network 0; le-services; les-bus x; list
```

Correct operation of the Redundancy VCC can be seen using the command sequence:

```
talk 5
network 0; le-services; les-bus x; show vccs
```

8.3.4 LES/BUS Peer Redundancy Use

LES/BUS Peer Redundancy is recommended for use in network designs where the location of the acting LES/BUS is not an issue.

In network designs where the location of the acting LES/BUS is a more important consideration than maximum network availability, Peer Redundancy should be disabled. One example might be the existence of a WAN connection between primary and backup LES/BUS. Another might be if tight control is needed on the distribution of workload between a set of MSS Servers in the steady state. In such environments where the location of the acting LES/BUS is critical, configure LES/BUS Redundancy with Peer Redundancy disabled because then the defined primary LES/BUS will be the acting LES/BUS whenever it is operational.

8.3.5 LES/BUS Peer Redundancy Migration Considerations

Peer Redundancy can operate only if all three of the following criteria are met:

1. Primary and backup LES/BUS are running MSS Server operational code containing Peer Redundancy support (MSS Release 2.2).
2. Primary LES/BUS has original Redundancy and Peer Redundancy enabled.
3. Backup LES/BUS has original Redundancy enabled.

Therefore, enabling Peer Redundancy on a primary LES/BUS would have no effect if the backup LES/BUS has a back-level version of MSS microcode. In this case, Peer Redundancy would only begin operation when the Backup LES/BUS microcode was upgraded to MSS Release 2.2. Note again that Peer Redundancy is disabled by default.

8.4 LECS Database Synchronization

MSS Release 2.2 includes a new feature, *LECS Database Synchronization,* to assist in the configuration of multiple LECS. It introduces the concept of a *Primary LECS*, which is used to control the configuration of all other LECSs, thus easing the configuration process.

Note

The LECS Database Synchronization feature is a *configuration assistant*. It will not automatically ensure all LECS are synchronized; this should be managed manually (see 8.4.4, “LECS Database Synchronization Use” on page 165).

8.4.1 Previous Implementation

Previous MSS releases did not include a LECS Database Synchronization feature. Without this feature the user must:

1. Manually configure all LECSs with the same parameters that control how LE Clients are configured and assigned to ELANs.
2. Manually ensure that any change to one LECS is mirrored in all other LECSs.

8.4.2 New Implementation

Using LECS Database Synchronization, the user need only configure ELANs on a single primary LECS. Each non-primary LECS is then updated by the primary. The non-primary LECS are referred to as *redundant*, although this is not an accurate description as all LECS can be in use simultaneously this term is used in the configuration process and will be used in this book.

The primary LECS keeps a list of all the redundant LECS. This list can be retrieved from the ATM switch using the Interim Local Management Interface (ILMI) or can be entered manually.

Having configured the primary LECS database with all ELANs, the user must send this database to the redundant ELANS. This can be achieved in one of three ways:

1. Issue the Talk 5 command - `write .`
2. Restart the primary LECS.
3. Reload the primary MSS.

When a redundant LECS receives a new LECS database from the primary, it will restart its LECS subsystem and start using the latest database. However, the received database is not automatically saved.This is because there may be other unsaved changes on the target MSS which the user does not want saved.

Note that the synchronization process uses the SRAM configuration on the primary LECS and not the *Talk 5* run-time configuration. Therefore, changes made to the primary LECS in *Talk 5 will not* be synchronized when the `write` command is used. Changes should be made in *Talk 6* or with the configuration tool, loaded into the primary MSS by reloading it and then sent.

The LECS Synchronization process is unidirectional; changes made to a redundant LECS list will not be sent to the primary.

There are some new *Talk 5* commands to enable the user to see the current status of synchronization (see 8.4.3.4 on page 165).

8.4.2.1 Details of the LECS Synchronization Process

LECS Database Synchronization from the primary LECS to all the redundant LECSs is performed serially to each of the redundant LECS instances in turn. The primary LECS will establish a point-to-point VCC called the *LECS Synchronization VCC.*

Setup of this LECS Synchronization VCC will be attempted only two times at most. If the initial VCC setup fails, the setup will be retried five seconds later. If the LECS Synchronization VCC is successfully established, the LECS databases

are synchronized and the VCC is disconnected. The primary LECS then repeats this process with the next LECS in the configured LECS Synchronization List.

The primary LECS performs the following steps:

1. Reads its SRAM records and encodes certain network information as user-defined Type/Length/Values (TLVs).
2. The encoded TLVs are appended to a LANE Configuration Request frame, called the LECS Synchronization Request Frame.
3. LECS Synchronization frames are formatted as LE Configuration frames and sent on the LECS Synchronization VCC.
4. If the frame is larger than the maximum Service Data Unit (SDU) size for the VCC, the LECS Synchronization Request Frame will be fragmented into multiple requests.

When a redundant LECS receives a Synchronization Request Frame it performs the following steps:

1. Decodes each user-defined TLV into a LECS SRAM structure.
2. All successfully decoded LECS Database Synchronization information received is saved to SRAM.
3. Information associated with TLVs that cannot be decoded or are erroneous is ignored and an error event is logged.
4. In the case of fragmented LECS Synchronization Request Frames, a status response indicating success or failure is returned after each Synchronization Request is processed.
5. A Synchronization Response Frame is sent by the redundant LECS over the LECS Synchronization VCC, to inform the primary of completion of LECS Synchronization processing.
6. After the Synchronized LECS database is written to SRAM, the redundant LECS restarts itself using the Talk 6 parameters.

8.4.3 LECS Database Synchronization Configuration

There are a number of new configuration options to allow LECS Database Synchronization to be configured for use.

8.4.3.1 Configuration Steps:

Primary LECS

1. Define a list of redundant LECS (configuring this list defines the LECS as primary).
   - Via ILMI (optional) and/or
   - Manually configured (optional)
2. Enable auto database synchronization (optional)
3. Configure all ELANS and policies in the normal way

Redundant LECS

1. Configure LECS as redundant

2. Allow remote LECS configuration (configuring this option defines the LECS as redundant, and allows the LECS database to be overridden by the primary)

8.4.3.2 Configuring Using the Configuration Tool

Figure 71 on page 162 and Figure 72 on page 162 show the navigation path and configuration window where a primary or redundant LECS is configured:

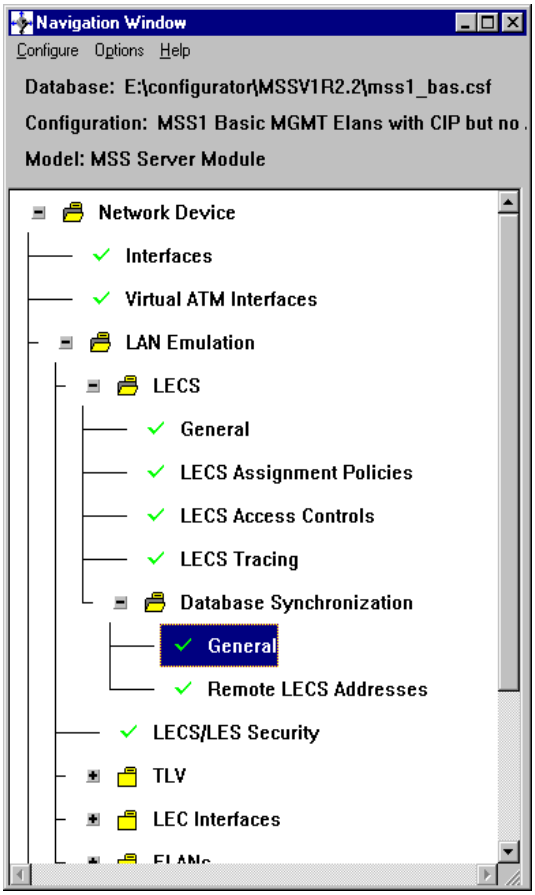


Figure 71. Navigation Path to Enable LECS Synchronization

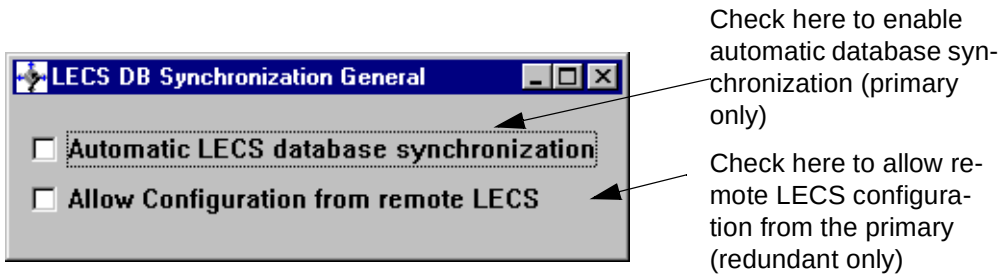


Figure 72. Configuration Window Enabling LECS Synchronization

Figure 73 on page 163 and Figure 74 on page 163 show the navigation path and window where manually entered redundant LECS addresses can be entered into the primary LECS:

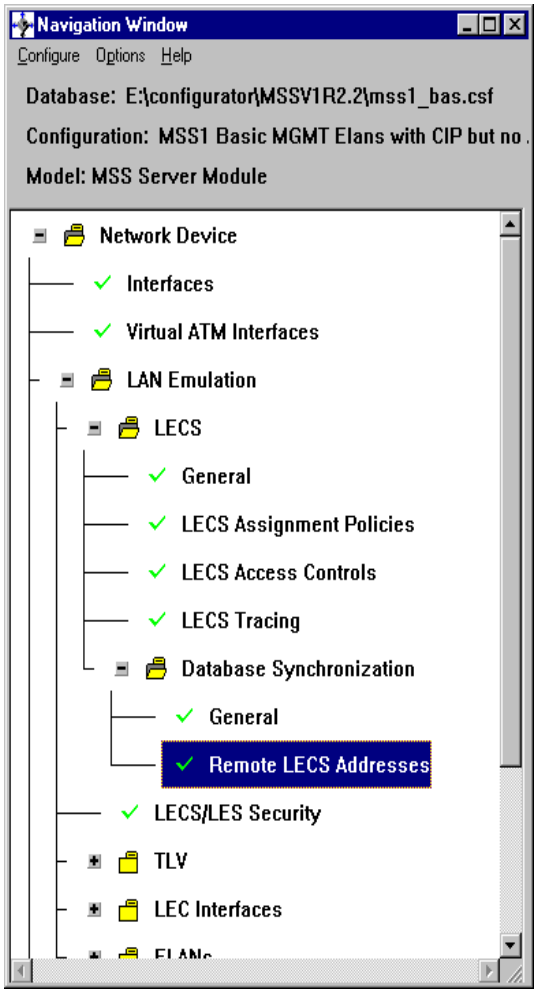
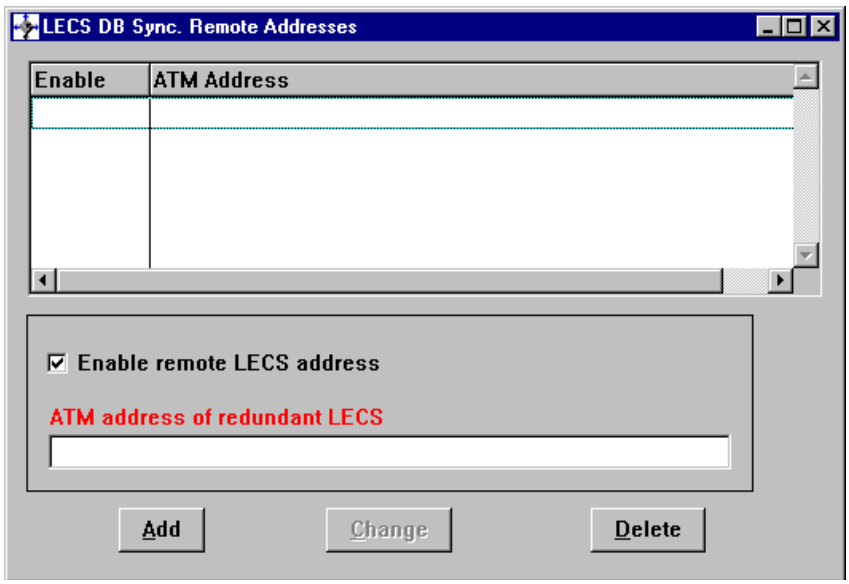


Figure 73. Navigation Path for Configuration of Remote LECS Addresses



Add the ATM address of each redundant LECS in this window

Figure 74. Configuration Window for Entering Addresses of Remote Redundant LECS

8.4.3.3 Configuring Using Talk 6

Figure 75 on page 164 shows the command line entries used to configure the Primary LECS:

```
MSS1 *TALK 6

MSS1 Config>NETWORK 0
ATM user configuration
MSS1 ATM Config>LE-SERVICES
LAN Emulation Services user configuration
MSS1 LE Services config>LECS
Lan Emulation Configuration Server configuration
MSS1 LECS config>ADD
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.01
End system identifier [1]?
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:
Standard ELS activated for LECS
MSS1 LECS config>SET AUTO
Synchronize LECS Databases Automatically [no]y       1
LECS Automatic Database Synchronization is Enabled
Selection "LECS Automatic Database Sync modification" Complete
MSS1 LECS config>DATABASE-Synchronization
LECS Database Synchronization configuration
MSS1 LECS Database Sync config>ADD-VIA-ILMI          2
Can't execute this option because:
No remote LECS ATM addresses retrieved via ILMI
MSS1 LECS Database Sync config>ADD-USER-DEFINED      3
Remote LECS ATM Address []? 40:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:0
0:00:00:00
Added remote LECS ATM address successfully
```

Figure 75. Primary LECS Configuration Using the Command Line

1 Enables automatic synchronization from this LECS. This is optional, and if not enabled then synchronization always has to be performed manually: see Figure 8.4.4.1 on page 166.

2 Retrieves the list of LECS using the ILMI process; this command is executed immediately.

3 Adds a user-defined LECS to the database (required if it is not possible to use ILMI or only specific LECS instances are to be defined).

Figure 76 shows the command line entries used to configure the Redundant LECS:

```
MSS1 *TALK 6

MSS1 Config>NETWORK 0
ATM user configuration
MSS1 ATM Config>LE-SERVICES
LAN Emulation Services user configuration
MSS1 LE Services config>LECS
Lan Emulation Configuration Server configuration
MSS1 LECS config>ADD
  (  1) Use burned in ESI
  (  2) 40.00.82.10.00.01
End system identifier [1]?
LECS added to configuration
Enable standard Error Logging System for LECS? [Yes]:
Standard ELS activated for LECS
MSS1 LECS config>SET REMOTE
Allow remote LECS configuration [no]y          1
LECS Remote Configuration is Enabled
Selection "Allow Remote LECS Configuration modification" Complete
```

Figure 76. Redundant LECS Configuration

1 This defines the LECS as redundant, and allows this LECS to be written to by the primary.

8.4.3.4 Confirming Correct Configuration

Using Talk 5 on the MSS with the primary LES/BUS, LECS Database Synchronization can be seen to be configured using the following command sequence:

```
talk 5
network 0; le-services; lecs; list
```

8.4.4 LECS Database Synchronization Use

LECS Database Synchronization is useful for networks which are prone to having a large number of changes made to their ELAN definitions, or for those which have a large number of LECS.

Synchronization can be configured to occur automatically each time the primary LECS is restarted, or manually using *Talk 5*.

Be aware of the following considerations:

- If a redundant LECS is down at the time of synchronization no later attempts will be made automatically, so when the redundant LECS becomes live again it will be running with an old version of the LECS.
- A redundant LECS will start using the newly synchronized LECS immediately, but will not save it in SRAM, therefore a subsequent reload of the LECS will result in it running an old version of the LECS.
- If more than one primary LECS has been defined in a network, a LECS database synchronized with one primary could be overridden by the other primary.

- Changes made in a redundant LECS will be overridden the next time a primary starts a synchronization.

To avoid situations in which the network is running with an unknown LECS setup due to the above considerations, we recommend the following:

- Use manual synchronization.
- Always check whether synchronization has been successful to every redundant LECS.
- Always save the configuration on every redundant LECS after successful synchronization.
- Ensure there is only one primary LECS in the network.
- Do not make manual changes to a redundant LECS database.

8.4.4.1 Manual Synchronization

The first step when initiating manual synchronization is to create a list of the LECS which you wish to update. This list will be a subset of any or all of the LECS which were discovered at ILMI time or entered manually during initial configuration.

1. Using the following *Talk 5* command sequence creates a subset of the LECS to be synchronized:

   ```
   talk 5
   network 0; le-services; lecs; database Synchronization; create
   ```

2. Using the following *Talk 5* command sequence initiates synchronization:

   ```
   network 0; le-services; lecs; database Synchronization; write
   ```

3. Using the following *Talk 5* command sequence checks the status of the synchronization process:

   ```
   talk 5
   network 0; le-services; lecs; database Synchronization; list
   ```

4. After successful synchronization with each redundant LECS, log on to the MSS Server providing the redundant LECS and save the configuration using *Talk 6*. Be aware that this will save *all* changes which might have been made to the configuration since the last reload, so first confirm that it is OK to save any other non-LECS-related changes which may have been made!

5. If synchronization to any LECS fails, investigate the cause and re-submit the write command for that LECS.

8.4.4.2 Automatic Synchronization

Automatic synchronization also requires a list of LECS to update. This subset is created in *Talk 6*.

1. Configure the primary LECS for automatic synchronization (see above).

2. Using the following *Talk 6* command sequence, enable a subset of the available LECS as candidates for automatic synchronization:

   ```
   talk 6
   network 0; le-services; lecs; database Synchronization; enable
   ```

8.4.5 LECS Database Synchronization Migration Considerations

LECS Database Synchronization is a proprietary IBM feature. It will only inter-operate will other IBM LECS running MSS Release 2.2 or higher. It is recommended that LECS Database Synchronization is used only in environments where all LECS are compatible.

8.5 Persistent Data Direct VCCs

MSS Server and MSS Client Release 2.2 includes a new feature, *Persistent Data Direct VCCs* (this applies to the MSS Server as well as the MSS Client; remember that the MSS Server itself has many LAN Emulation Clients). This feature is for use in networks where redundant LES/BUS is used. Its purpose is to reduce the network outage which occurs as a result of a primary LES/BUS failure. In addition to the Persistent Data Direct VCC feature, there are two non-configurable enhancements to the LEC code on the MSS Server and MSS Client which also help reduce network outage: *Rapid LES/BUS Failure Detection* and *Multiple LECS Configuration Requests*. These two features increase the speed at which a LEC will reconnect to the backup LES/BUS and are also discussed in this section.

Other LAN Emulation Clients

We should note that other LAN Emulation Clients are also capable of implementing Persistent Data Direct VCCs, specifically many of the LAN Emulation Clients that are implemented as device drivers in PC ATM LAN Adapters. This section of the book is talking about MSS's specific implementation of this function, but much of the discussion, if not the details of the implementation, will also apply to *these* LAN Emulation Clients.

8.5.1 Previous Implementation

Without this feature, a failure of the primary LES/BUS would result in all the VCCs for that ELAN being disconnected. The user would therefore experience a network outage.

When a LEC that was connected to this ELAN notices the control VCCs have been disconnected it will return to the LECS to determine the ATM address of a usable LES/BUS. If the LECS is not co-resident with the LES/BUSs for this ELAN it will not know that the primary LES/BUS has failed. The LECS will therefore give the LEC the address of the failed primary LES/BUS. The LEC will then make an abortive attempt to connect to the failed primary LES/BUS, it will then re-initialise itself and return to the LECS which will then provide the address of the backup LES/BUS. The LEC re-initialisation may take many seconds.

Once the LEC has joined the backup LES/BUS, all the control VCCs are re-established and connectivity will return.

8.5.2 New Implementation

8.5.2.1 Multiple LECS Configuration Requests

In order to avoid the time taken for the LEC to re-initialise while searching for an active LES/BUS, MSS Server and MSS Client Release 2.2 LEC code will allow

the LEC to return to the LECS twice before re-initialising itself. This feature has no configuration and is always enabled.

8.5.2.2 Persistent Data Direct VCCs

This feature brings the greatest advantage because in most cases it allows user traffic to continue after a LES/BUS failure so that the user will experience no network outage. With the *Persistent Data Direct VCC Mode* enabled, if the LEC loses its connection to the LES/BUS then the data direct VCCs are not immediately dropped. Instead, the following new actions occur:

- The LE_ARP timer is stopped.
- The LEC Reconnect Time-out timer is started.
- While the LEC is not joined to a LES/BUS, the LEC does not LE_ARP for any new target LAN destinations and does not forward unknown unicast or broadcast/multicast traffic.
- Provided that both LECs at each end of a data direct VCC support this feature, known unicast traffic (LE_ARP entry exists for target LAN destination) will continue to be transmitted over existing data direct VCCs.

The LEC returns to the LECS to obtain the backup LES ATM address and attempts to rejoin an ELAN. Once the LEC rejoins an ELAN, the LE_ARP timer is restarted and the LEC Reconnect Timer is stopped. The newly assigned LEC ID replaces the previous LEC ID. The LEC can rejoin an ELAN quickly, so no disruption in directed network traffic flows should occur. All data direct VCCs are disconnected after the Reconnect Time-out time expires if the LEC fails to rejoin an ELAN.

8.5.3 Persistent Data Direct VCCs Configuration

There are two new configuration options to allow the configuration of Persistent Data Direct VCCs. The following steps should be configured for each LEC on the MSS Server or MSS Client for which Persistent Data Direct VCCs are required.

8.5.3.1 Configuration Steps

1. Create the LEC in the normal way.
2. Enable *Data Direct Mode* for this LEC (disabled by default).
3. Modify the *Data Direct Time-out* if required, 10 to 300 seconds (30 by default).

8.5.3.2 Configuring Using the Configuration Tool

Figure 77 on page 169 and Figure 78 on page 169 show the navigation path and configuration program window for configuration of Persistent Data Direct VCCs.

Note

The Configuration Tool for the MSS Family Client Release 2.2 ptf 0 has no option for configuring Persistent Data Direct VCCs. Use *Talk 6* to configure this parameter on the MSSC. This option will be added in later ptf versions of the tool.

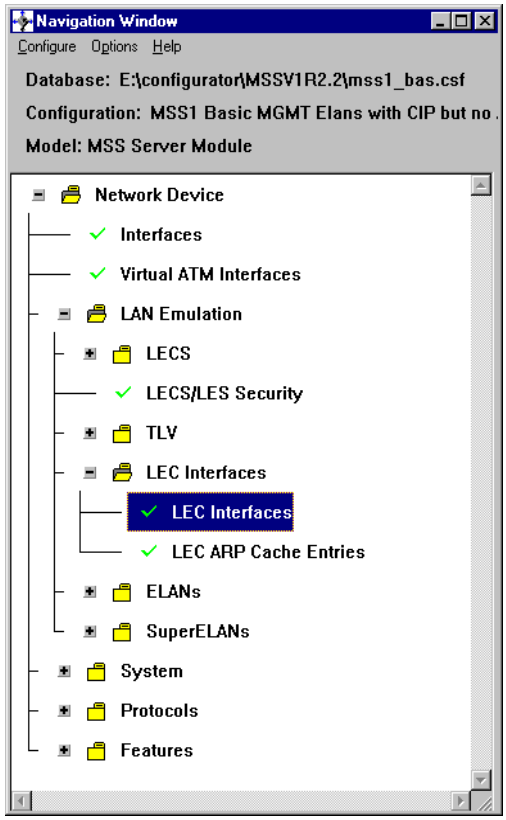


Figure 77. Navigation Path for Persistent Data Direct VCCs

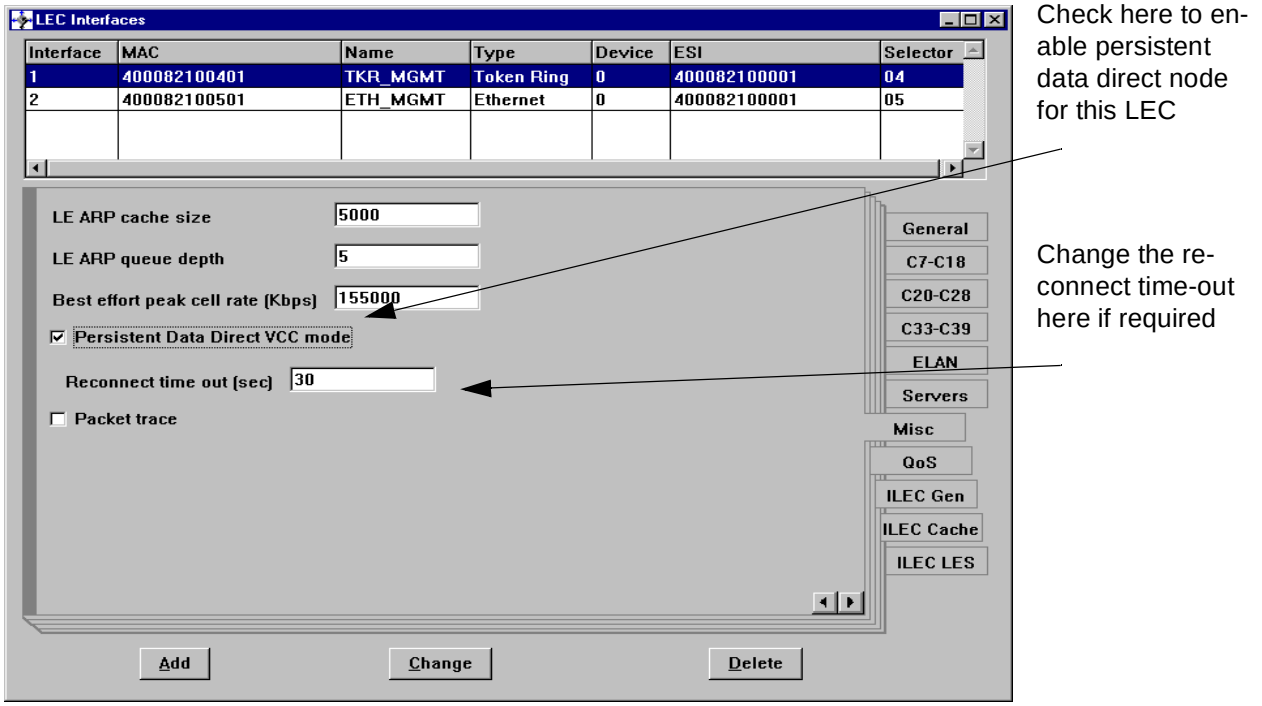


Figure 78. Configuration Window for Persistent Data Direct VCCs

8.5.3.3 Configuring Using Talk 6

Figure 79 shows the command line entries used to configure Persistent Data Direct VCCs:

```
MSS1 *TALK 6

MSS1 Config>NETWORK 0
ATM user configuration
MSS1 ATM Config>LE-CLIENT
ATM LAN Emulation Clients configuration
MSS1 LE Client config>CONFIG
Emulated LAN interface number [1]? ?
1     : ATM Token Ring LAN Emulation: TKR_1
2     : ATM Token Ring LAN Emulation: TKR_2
3     : ATM Token Ring LAN Emulation: TKR_PROD
4     : ATM Ethernet LAN Emulation: ETH_PROD
Emulated LAN interface number [1]? 1
ATM LAN Emulation Client configuration
MSS1 Token Ring Forum Compliant LEC Config>SET DATA-DIRECT-VCC-MODE
Enable data direct VCC mode? [Yes]:    1
MSS1 Token Ring Forum Compliant LEC Config>SET DATA-DIRECT-TIMEOUT
Data direct timeout [30]?              2
```

Figure 79. LE Client Configuration

1 Enables Persistent Data Direct VCCs for this LEC.

2 A different Data Direct Time-out can be entered here.

8.5.4 Use of Persistent Data Direct VCCs

The Persistent Data Direct feature should be enabled whenever a redundant LES/BUS is used. The default LEC Reconnect Time-out timer value of 30 seconds should be sufficient in most cases, but may be extended if necessary. The following points should be considered:

- Only Data Direct VCCs are maintained during the period when the LEC is not connected to a LES/BUS. Therefore multicast and broadcast packets will be discarded. Higher level protocols that depend on this type of traffic will notice disruption.
- In the unusual situation in which two LE Clients from a single failed LES/BUS are re-connected to two separate LES/BUSs, there will be a short period of time when a data direct VCC will connect two different ELANs. This may result in a network loop causing degraded network performance, but these VCCs will be disconnected as LE_ARP entries are verified.
- The LEC at both ends of the Data Direct VCC must support this feature in order for the VCC to remain active.

8.5.5 Persistent Data Direct VCCs Migration Considerations

This enhancement is independent of the LES/BUS implementation. If the LECs at both ends of a Data Direct VCC support this function the benefits will be realised. The feature is compatible with LE Clients that do not support this function; the VCC will then be treated in the normal way and disconnected during the LES/BUS failure. The implementation of this feature is proprietary to IBM, although presumably possible to implement in other devices.

8.6 Conclusion

Resilience is playing an ever increasing role in network design as users demand greater availability from the network and customers are willing to pay for additional hardware that increases the availability of the network even if it isn't normally used.

This release builds on the already rich resilience feature set of the MSS. For a listing of all resilience features and at which release level they became available see 4.2, "MSS Family History (Software)" on page 74.

LES/BUS Enhanced Redundancy can reduce the time taken for a backup LES/BUS to take over from a failed primary. It requires no configuration and is fully compatible with previous releases.

LES/BUS Peer Redundancy can eliminate a network outage caused by bringing a failed primary LES/BUS back into service. A new configuration parameter is used to enable this function.

LECS Database Synchronization can make it easier to configure complex networks with multiple LECS by enabling the network administrator to configure a single LECS and then copy this to any number of others.

Persistent Data Direct VCCs may have the biggest impact on improving network availability by enabling LE Clients to continue passing user data while a backup LES/BUS is in the process of taking over from a failed primary.

Chapter 9. MSS V2.2 MPOA Enhancements

This chapter lists the new features and enhancements which specifically affect MPOA within the MSS Server and MSS Client. Where appropriate *real life* examples will be given.

9.1 Implementing MPOA for IPX

MSS Release 2.2 expands the support provided in MSS Release 2.1 for Multiprotocol over ATM by adding support for the IPX protocol.

The MPOA specification[1] standardises the virtual router model allowing Layer 3 protocols such as IP and IPX to establish intersubnet (IP) and internetwork (IPX) shortcuts for efficient utilisation of resources in an ATM network. MPOA works in the LAN Emulation over ATM environment specified by the ATM Forum and uses the IETF's Next Hop Routing Protocol (NHRP) standard (RFC 2332) for interserver communication.

The previous implementation of MPOA in MSS Release 2.1 supported only the IP protocol. With MSS Release 2.2 support for IPX MPOA Client and Server is added. As with the previous version, the MPOA Client implementation for IPX lies in the MSS Client, while the MPOA Server implementation resides in the MSS Server.

This chapter addresses the additions and changes necessary to support the IPX protocol in MSS Release 2.2. Configuration examples are given for both MPOA Client and MPOA Server for IPX.

The following topics will be covered in this chapter:

- MPOA Server for IPX
- MPOA Client for IPX
- MPOA for IPX configuration examples
- MPOA Server MIB
- MPOA Client MIB

At this stage in the book, the reader is assumed to be familiar with MPOA basics and functions, and with the use and configuration of MPOA for IP. An introduction to MPOA and its benefits is given in this document in 1.2.2, “The MPOA Standard” on page 20. Details of the previous IP implementation of MPOA are given in *MSS Release 2.1, Including the MSS Client and Domain Client,* SG24-5231.

9.1.1 MPOA Server for IPX

The key benefits of MPOA are scalability and manageability. The MPOA Server (MPS) performs route calculations and helps to establish shortcuts between MPOA Clients (MPCs) which are used for high-speed Layer 2 hardware forwarding across the ATM switched network. Distributing the route calculation and forwarding functions allows for the incremental addition of forwarding capacity by adding more MPCs and overall throughput by increasing ATM network capacity.

[1] ATM Forum, Multiprotocol over ATM Version 1.0, AF-MPOA-0087.000, July 1997

The following items apply to the MPOA for IPX implementation in MSS 2.2:

- Fully compliant with the ATM Forum's MPOA specification
- Provides manageability and scalability benefits of the virtual router
- Supports the optional network aggregation feature, which can offer significant scalability advantages
- Complete auto-configuration support with no specific MPC configuration required, and the capability of receiving configuration parameters from the LAN Emulation Configuration Server (LECS)
- Supports CBR, rt-VBR, and nrt-VBR ATM Service Categories in addition to best-effort service with configurable traffic parameters
- Extensive manageability capabilities including several message logging categories, packet tracing, and availability of all information specified in the MPOA MIB
- Ability to independently enable/disable IP and IPX

The Release 2.2 MPOA Server provides support for setting up shortcuts for IPX traffic between MPOA Clients. The NHRP implementation for the MSS Server has been upgraded to support transit NHS function for IPX control flows[2]. However, there is no support for a NHRP Client for IPX in MSS; NHRP Client support remains only for IP shortcuts between IP stations. Unlike the implementation of MPOA for IP, MPOA for IPX does not support the establishment of shortcuts to NHCs and non-MPC LANE Clients.

IPX implementation for MPOA is derived from IP implementation for MPOA so that it is very similar to it in configuration and implementation.

Figure 80 on page 175 depicts a typical network with MPOA for IPX enabled. The MPS operation for IPX is completely equivalent to that for IP. An example of MPOA control flows is shown in Figure 81 on page 176.

When MPC1 detects that traffic to a particular destination exceeds the configured threshold (the default is 10 packets/sec.), it initiates shortcut establishment by sending an MPOA Resolution Request to the MPOA Server (which is an IPX router, MPS1). The MPOA Server receives the request from an MPOA Client asking for the information required to set up a shortcut to an IPX destination. The MPOA Server either replies with the ATM address of the MPOA Client that is serving this destination (if it knows it) or the MPOA Server forwards the request (using NHRP control flows) along the default routed path to another MPOA Server. In our case, MPS2 detects an MPC in the path towards the destination and issues a Cache Imposition Request to the egress MPC. Given a successful response from the MPOA Client, MPS2 sends an NHRP Resolution Reply to MPS1 which is in turn transformed into an MPOA Resolution Reply and sent back to MPC1. MPC1 now has all the information that it requires to establish a shortcut VCC to MPC2, and sends traffic through it once it is set up. MPC2 uses the data link (DLL) header information provided in the Cache Imposition Request to send the traffic received over the shortcut VCC onwards to its destination, making it appear to the final host as if had actually still used the default path shown in Figure 80 on page 175.

[2] This initially seems strange but is required because MPOA uses NHRP as the discovery protocol between multiple MPOA Servers.

If MPS2 detects a routing/bridging change to a destination for which a shortcut is in use, it invalidates the shortcut by sending an MPOA Cache Imposition (Purge) to MPC2 and sending an NHRP Purge Request to MPS1. MPS1 then is responsible for sending an NHRP Purge Request to MPC1 which will cause the ingress MPOA Client to remove its cache entry. Note that the diagram shows, correctly, that this control flow is actually an NHRP Purge Request; the MPOA standard obviously decided that there was no requirement for an “MPOA Purge Request” when the existing architecture was adequate for the purpose.

For a complete description of MPC-to-MPC flows, refer to *The ATM Forum Technical Committee, Multiprotocol over ATM Version 1.0,* AF-MPOA-0087.000, July 1997, since the architected flows are equally applicable to both IPX and IP.

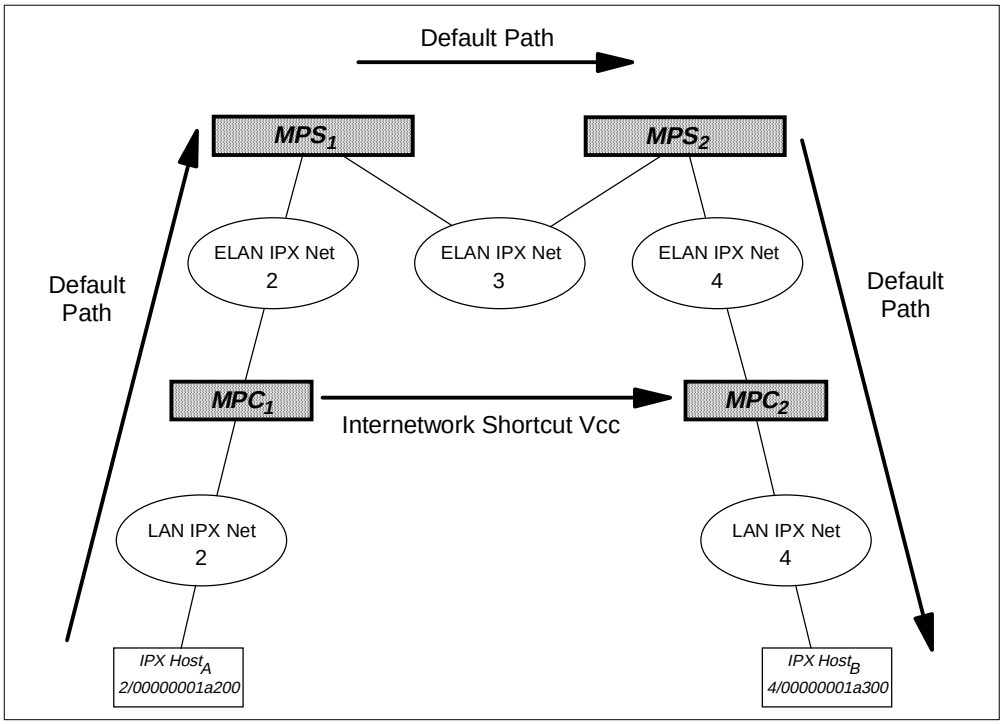


Figure 80. MPOA for IPX Scenario

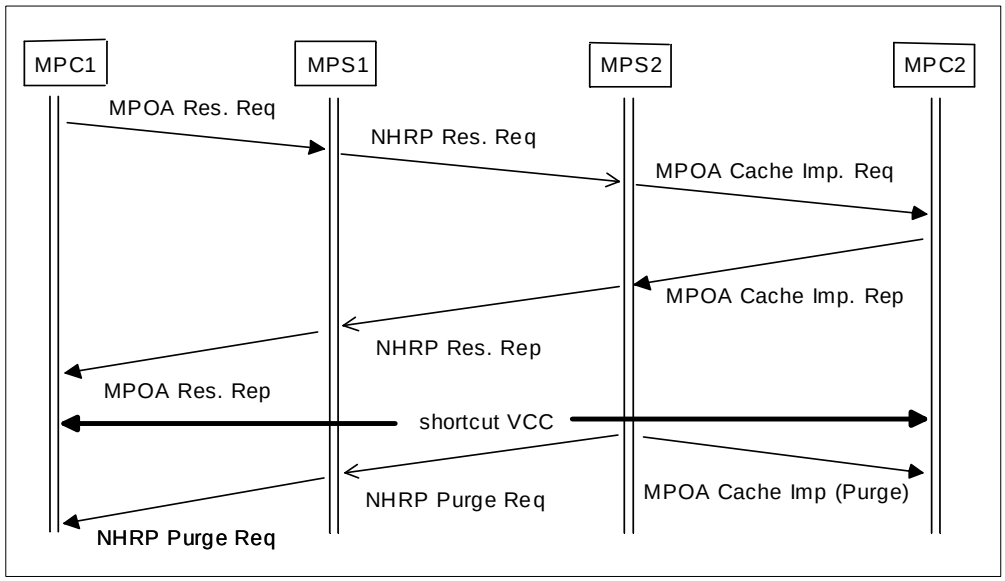


Figure 81. MPC-to-MPC Control Flows for IPX shortcuts

9.1.2 MPOA Client for IPX

To complete the MPOA support for IPX, the MPOA Client for IPX is available in MSS Client Release 2.2.

The MPOA Client for IPX is fully compliant with the ATM Forum's MPOA specification and performs almost equivalently to the MPOA Client for IP.

As stated above, the MPOA *Server* current implementation does not support the establishment of shortcuts between an IPX MPOA Client and either NHRP Clients or non-MPC LANE Clients. However, the MPOA *Client* does support shortcuts for IPX traffic to these other types of non-MPOA Clients provided that it uses a server which also supports these; since no such servers exist it only currently uses the ability to use LANE Encapsulation as a method of boosting performance over shortcuts to other MPOA Clients.

Shortcuts between different media types are supported. At the moment of writing this book, MPOA Client support for token-ring was performed by the MSS Client and the equivalent function for Ethernet networks is available in the 8371 Multilayer Ethernet Switch. IP and IPX traffic is supported in both cases.

A basic setup and flow of control frames for IPX MPOA has already been depicted in the previous section. Only special aspects that apply to the MPOA Client will be mentioned here.

When a shortcut is established between two MPC Clients, it is always in one direction. For traffic in the opposite direction to travel on a shortcut, the same process for establishing an MPOA shortcut has to be repeated in the reverse direction. The other MPOA Client is the one responsible for initiating this Resolution Request, which will occur only if the traffic rate exceeds *this* Client's configured threshold value. However, if such a reverse shortcut is established, the two shortcuts have the potential to share the same VCC. If the two shortcuts are established at roughly the same time, it could well happen that each one initially uses a separate VCC. After a time-out period, however, both shortcuts will switch

to using the same VCC (normally the VCC initiated by the lower ATM address MPC) and the other VCC will age out after a further short while and be released.

Established shortcuts are only valid for a time interval specified by the MPS in a Resolution Reply or an Imposition Request. They must therefore be *refreshed* periodically by initiating another Resolution Request process. Established shortcuts are also monitored to ensure that they remain in use and are purged if not. The timings for refreshing and purging shortcuts are the same for IPX as for IP and are configurable in the MPS.

An IBM extension to the standard allows the establishment of shortcuts to LAN Emulation Clients that are not running as MPOA Clients. These shortcuts are called LANE-encapsulated shortcuts as the data is encapsulated in the same format as normally flows for LAN emulation. This encapsulation allows IP MPOA Clients to establish shortcuts to non-MPC LANE stations, but this method is also used by MSS Clients MPCs to take advantage of the high-performance hardware switching capabilities of the MSS Client platform. See 1.3.2, “LANE Encapsulation and Vendor Extensions” on page 27 for additional information on this topic.

As stated above, MSS Release 2.2 MPOA Clients for IPX have support for these LANE shortcuts. The current implementation of the MPOA Server does not allow the establishment of shortcuts to LANE Clients for the transport of IPX traffic, but it does allow two MPOA Clients to establish LANE shortcuts between them, even for IPX traffic, which allows all the packets passing through the egress MPOA Client to be switched in the hardware of the ATM edge device.

When the shortcut is established between a token-ring MSS Client and an 8371 Multilayer Ethernet Switch, the encapsulation used is different for each direction. Traffic from the MSS Client to the 8371 uses a shortcut with encapsulation format 1483 Tagged. Traffic from the 8371 to the MSS Client uses LANE encapsulation. This is done to best use the resources of both machines and provide the fastest path for data packets in both directions. See 1.3.1, ““Tagged” and “Non-Tagged” Frame Formats” on page 26 for additional information on this topic.

Despite similarities, there are some differences between the IPX implementation of MPOA and that of IP:

- IPX uses a transport control (TC) field which is incremented as a packet moves through a network, as opposed to IP which decrements a time to live (TTL) field. An ingress MPC increments the TC count before sending an IPX packet on a shortcut. If the TC count has reached the maximum value specified in the hop count extension value in the associated Resolution Reply, the packet is forwarded unmodified to the MPS via LANE.
- There is no fragmentation in IPX. It is done automatically by higher layers at the end hosts.
- There are four types of data link encapsulation supported for IPX over Ethernet as shown in Figure 82 on page 178. MPC support for destinations with any of these encapsulation types is provided in Release 2.2. There are also two types of data link encapsulation for IPX over token-ring, as shown in Figure 83 on page 178. They are both supported as well.

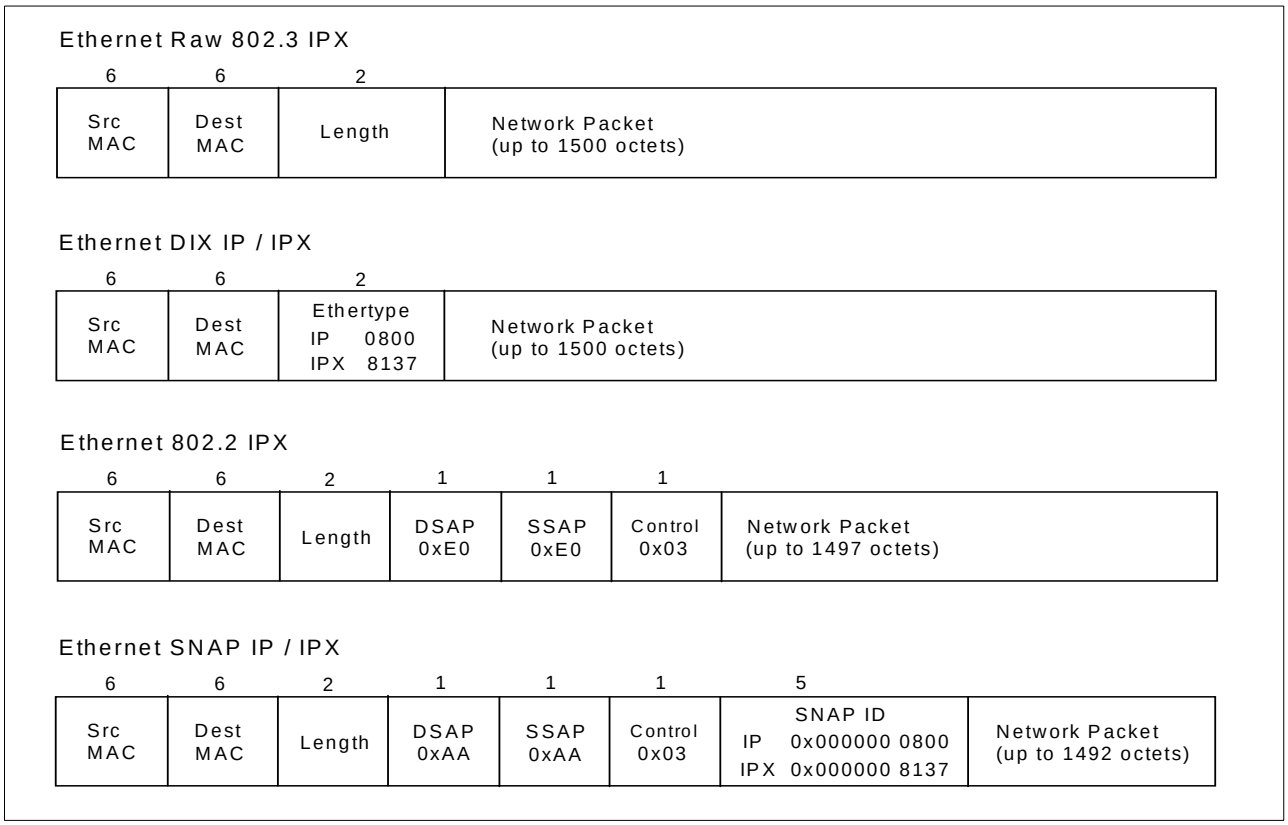


Figure 82. Ethernet Encapsulation for IPX

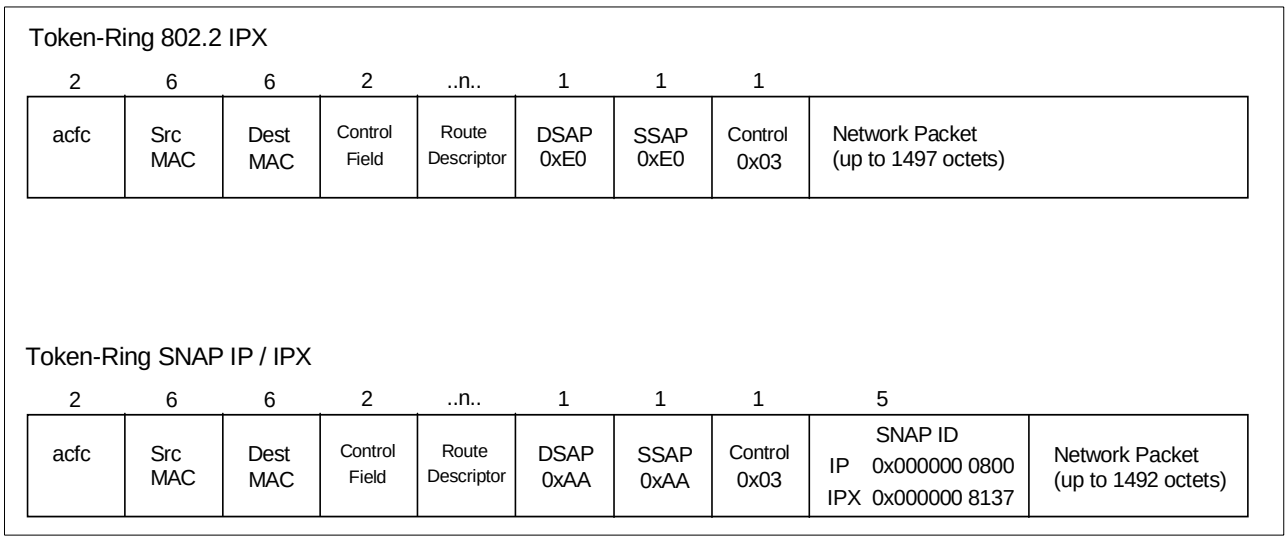


Figure 83. Token-ring Encapsulation for IPX

It is worth mentioning the following points about the way the MSS Release 2.2 handles some specific IPX packets:

- All IPX packet types, except Propagated Packets (type = 0x14) and packets with special destinations (as explained later), are passed over shortcuts.

Propagated Packets are always forwarded along the routed path so that the IPX routers can process them correctly.

- In IPX, an “all ones” destination node address is considered to be a broadcast to all nodes on the destination network. Therefore, any packet with an “all ones” destination node address will be forwarded along the routed path to ensure that it is received by all nodes in the destination network. Note that Propagated Packets (type=0x14) should normally have “all ones” for the destination node address. Only unicast packets are sent over shortcuts.
- In IPX, a packet with an “all zeros” destination network address is considered to be destined for the current network. Therefore, any packet with an “all zeros” destination network address will be forwarded along the routed path to ensure that it is received by all the nodes in the current network.

Great effort has been made to keep the performance of the MPOA Client as high as possible and, in particular, great care has been taken so that the performance of the MPOA Client for IP was not reduced by adding support for IPX. This impacts the way that the code has been implemented.

9.2 MPOA for IPX Configuration and Use

MPOA for IPX is configured in just the same as MPOA for IP. Minor changes have been made to the configuration menus and tool so that IPX configuration is possible. Note that the configuration program’s panels have been rearranged so that MPOA and NHRP menus are no longer a subset of *IP configuration* but instead have their own separate configuration section.

MPOA for IP and IPX can be enabled independently. *Exclude Lists* and *Disallowed Rtr-Rtr Shortcuts* can be defined separately for IP and for IPX. The configuration parameters in both cases are similar: for IP the IP address and IP mask are configured; for IPX, net number and node number are required instead.

The rest of the MPOA parameters are configured globally and are not specific to IP or IPX protocols.

Some of the configuration parameters can be changed dynamically in *Talk 5*. On the MPOA Client it is possible to change parameters using *Talk 6* and reload them to the active configuration without rebooting the MSS Client itself. More information on this is given in the following sections. Note that it is not generally possible to configure a protocol using *Talk 5* which was not previously enabled in *Talk 6* at the time the MSS was started; this is certainly true for IPX and for MPOA.

As was the case for IP, all MPOA Clients can take their configuration information from the LAN Emulation Configuration Server (LECS). MPOA Clients and MPOA Servers discover each other during the normal flow of packets through the routed path. This makes a plug-and-play environment for MPOA for IP and IPX possible.

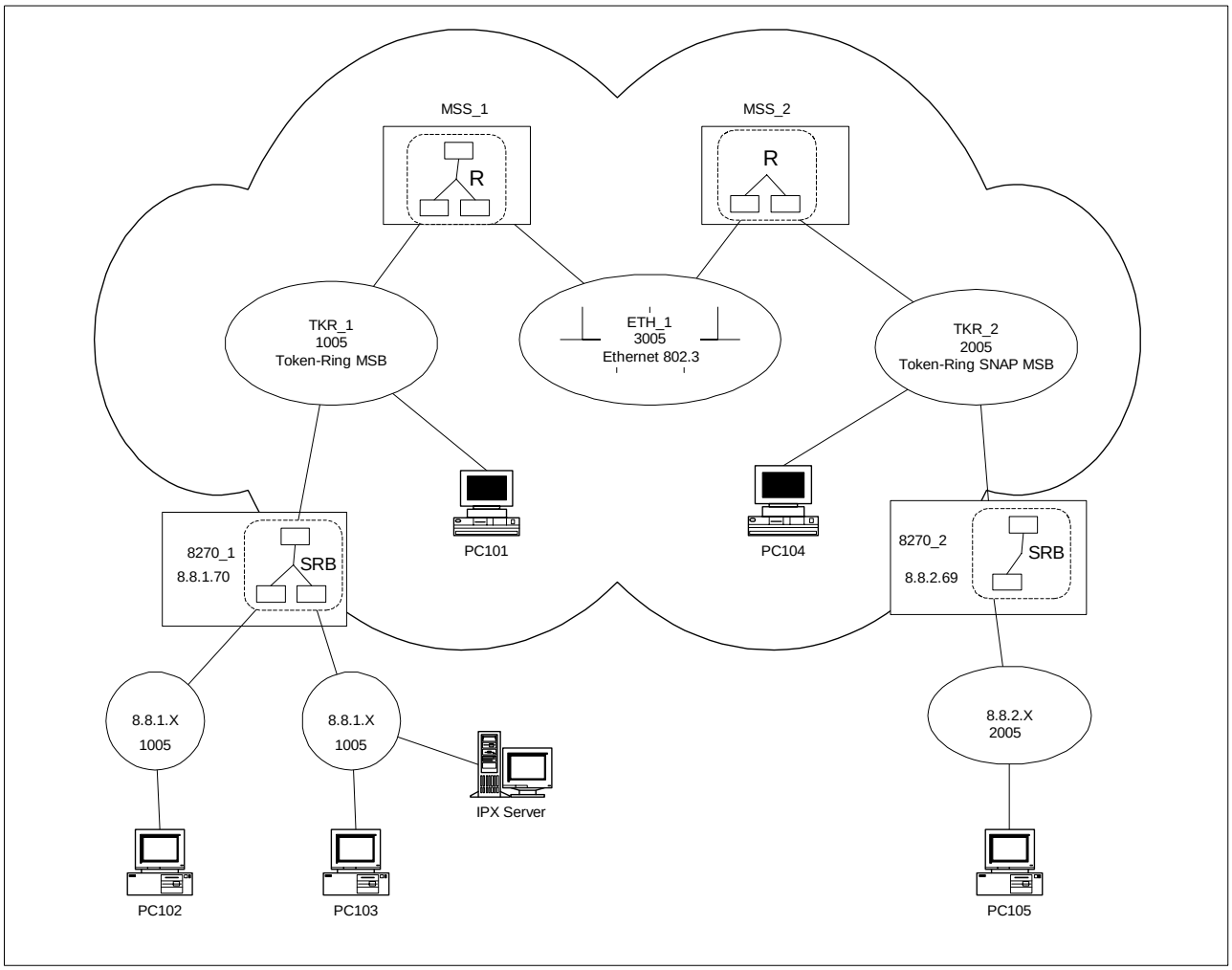


Figure 84. MPOA for IPX Example Setup

To show how to configure MPOA for IPX we have set up an example network, depicted in Figure 84. In this network we have two MSS Servers, each configured for IPX Routing and as MPOA Servers for IPX. Two 8270s with one MSS Client each are configured for source-route bridging and have the MPOA Client for IPX activated. There are six IPX devices in total: two IPX clients running on ATM-attached stations configured to use LANE, three IPX clients running on token-ring networks and one IPX Server on a token-ring running Netware 4.1.0. Some of the clients are running Windows 95 and some of the clients are running Windows NT clients.

Since the IPX server and IPX clients have to talk across a source-routed network, source-route support for IPX has to be loaded on all machines. In the IPX server this is done by loading the ROUTE.NLM driver and binding it to its network adapter. In the Windows 95 clients, source-route support is enabled by clicking the appropriate parameter in the IPX/SPX protocol configuration window. In the Windows NT stations, source-route support is enabled by default[3].

[3] In fact to disable it, the setting for **SourceRouting** in the Registry has to be modified. Further discussion of the implementation of IPX on Windows NT clients is outside the scope of this book!

9.2.1 MPOA Server for IPX Configuration

To configure the MPOA Server for IPX, the MSS Server first of all needs to be configured as an IPX router.

These are the steps that have to be followed (the first four steps are really just the steps required to configure a normal IPX router in the MSS Server):

1. Create interfaces (LECs) in all the ELANs that MSS will be routing between
2. Enable IPX at box level
3. Add IPX network addresses to all the interfaces that are needed, at least one address to each interface created in step (1)
4. Add IPX Static Routes if needed or set up an appropriate dynamic routing protocol such as RIP
5. Enable the MPOA Server for IPX globally for the MSS Server
6. Enable the MPOA Server for each interface that will require to use it
7. Configure global MPOA Server parameters
8. Define, if needed, the *Exclude List* and *Disallowed Rtr-Rtr* shortcuts list for IPX

At this point the reader is assumed to be familiar with basic MSS Server configuration and in particular with the definition of LES/BUS, LECS and LECs, and in how to configure routing and bridging. If not, for more information on these functions and how to configure them in the MSS, please refer to Appendix D, “Related Publications” on page 221.

Either the configuration tool or the command line can be used to configure MPOA for IPX. If either *Talk 6* or the configuration program is used then the MSS has to be rebooted to activate new changes. When using *Talk 5*, the new changes take effect immediately but cannot be saved and are lost at the next reboot of the machine.

Not all the parameters can be configured from *Talk 5*. In particular, adding elements to the Exclude List or the Disallowed Rtr-Rtr Shortcut List is not supported, nor is it possible to enable/disable the MPOA Server for any protocol or to configure any interface individually: only the global MPOA Server parameters can be changed this way.

This may seem a little inflexible; it is possible to change many of the above parameters using *Talk 6* and then use the RESET command under *Talk 5.* This restarts the MPOA Server service with the new parameters that are in the SRAM without the need to reset the entire MSS Server.

9.2.1.1 Setting the MSS as an IPX Router

To define routing for IPX in the MSS it is simply a case of enabling IPX globally and of then configuring IPX addresses on the required interfaces.

To do so, *IPX Circuits* have to be defined for each interface. This can only be done using *Talk 6* or the configuration tool; subsequently Circuits can be individually enabled or disabled using *Talk 5*. The RESET command in *Talk 5* allows the IPX configuration parameters to be reloaded to the latest version modified in *Talk 6* and now residing in SRAM.

To define an IPX Circuit use the *Talk 6* command `ADD BROADCAST-CIRCUIT`. The system will prompt for an interface number, circuit number and network number. Then select the IPX frame type with the command `FRAME`.

To set up IPX routing between IPX routers it is necessary to enable RIP or to add static routes. Since RIP is enabled by default on all circuits there is actually no need to make any specific configuration steps if this is what is desired.

As for the rest of the IPX parameters, the default values usually work well unless *Access controls* or other additional features need to be enabled. In Figure 85 on page 182 and Figure 86 on page 183 we show the configuration for MSS_1 using the configuration tool. In Figure 87 on page 184 we show the command-line equivalent used to configure MSS_2 in our network.

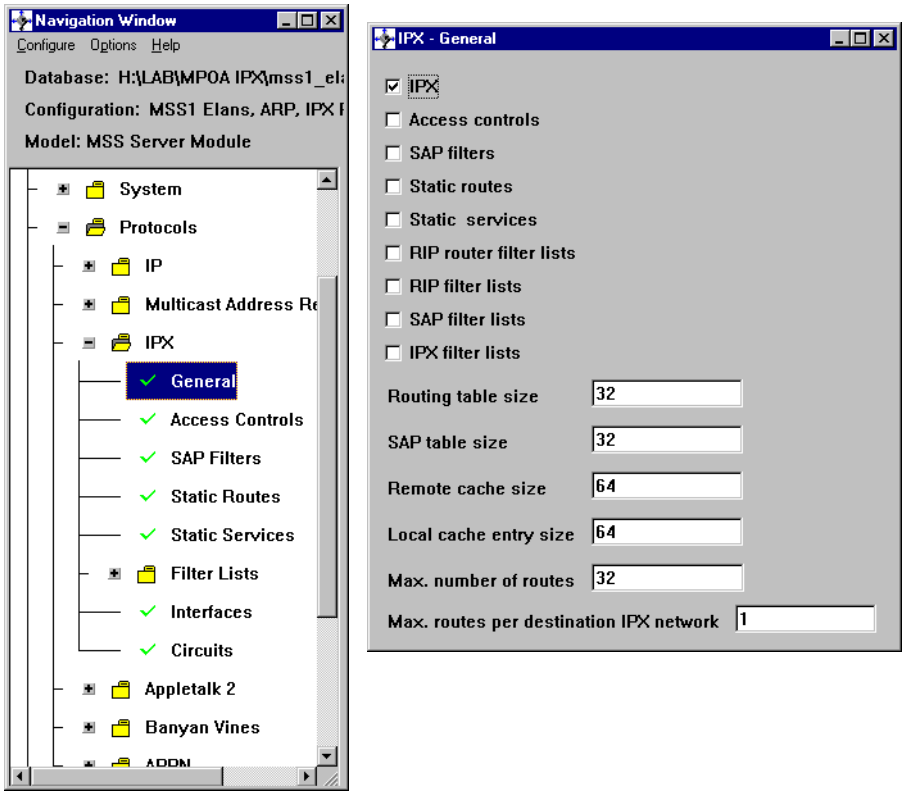


Figure 85. IPX General Configuration for MSS_1 Using the Configuration Program

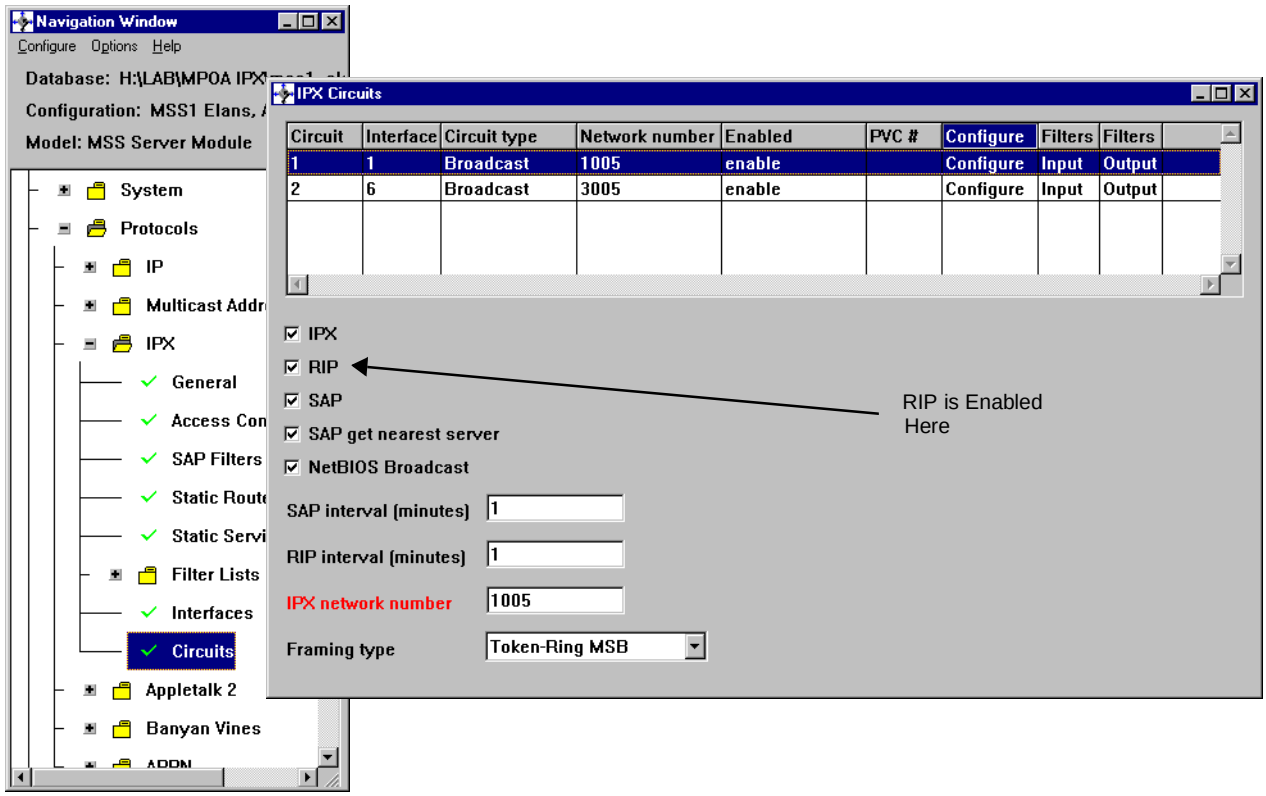


Figure 86. IPX Interface Configuration Using the Configuration Program

```
MSS2 *TALK 6

MSS2 Config>NETWORK ?
0     : ATM
1     : ATM Token Ring LAN Emulation: TKR_2
2     : ATM Ethernet LAN Emulation: ETH_1
3     : ATM Token Ring LAN Emulation: TKR_PROD
Network number [0]? Q
Invalid network number
MSS2 Config>PROTOCOL ipx
IPX protocol user configuration
MSS2 IPX config>ENABLE IPX
MSS2 IPX config>ADD BROADCAST-CIRCUIT
Which interface [0]? 1
IPX circuit number [1]?
IPX network number in hex
('0' is only allowed on IPXWAN unnumbered circuits) [1]? 2005
MSS2 IPX config>ADD BROADCAST-CIRCUIT
Which interface [0]? 2
IPX circuit number [2]?
IPX network number in hex
('0' is only allowed on IPXWAN unnumbered circuits) [1]? 3005
MSS2 IPX config>FRAME TOKEN-RING_SNAP MSB
IPX circuit number [1]?
MSS2 IPX config>FRAME ETHERNET_8023
IPX circuit number [1]? 2
MSS2 IPX config>LIST ALL
IPX Globals:
------------
IPX Globally                Enabled
Host Number (serial line)   000000000000
Maximum Services                 32
Maximum Networks                 32
Maximum Routes                   32
Maximum Routes per Destination   1
Maximum Local Cache entries      64
Maximum Remote Cache entries     64
Keepalive-Filtering Table Size   32

IPX Configuration:
------------------
                                 NetBIOS    Keepalive
Circ   Ifc    NetNum    IPX      Broadcast  Filtering  Encapsulation
1      1      2005      Enabled  Enabled    Disabled   TOKEN-RING_SNAP MSB
2      2      3005      Enabled  Enabled    Disabled   ETHERNET_802.3

RIP Configuration:
------------------
                                 Update    Split      Broadcast
Circ   Ifc    NetNum    RIP      Interval  Horizon    Pacing
--More--
MSS2 IPX config>
```

Figure 87. Configuration of IPX Routing for MSS_2 Using the Command Line Interface

9.2.1.2 Enabling and Configuring MPOA for IPX

Once IPX routing is defined, the next step is to enable MPOA for IPX at box level and on the interfaces required (note that by default MPOA for IPX is enabled on all interfaces once it has been enabled globally). No other special configuration is needed since default values work well. Our only recommendation is to change the locally-configured end system identifier (ESI) for the MPS, for administrative purposes.

Figure 88 and Figure 89 show the steps used to configure MPOA for IPX on MSS_1 using the configuration program:

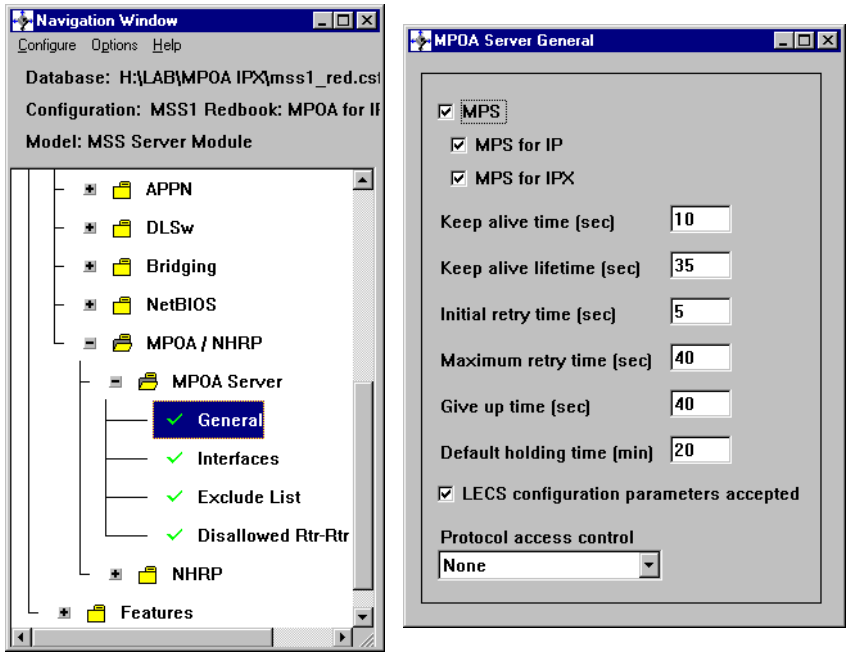


Figure 88. MPOA for IPX General Configuration for MSS_1

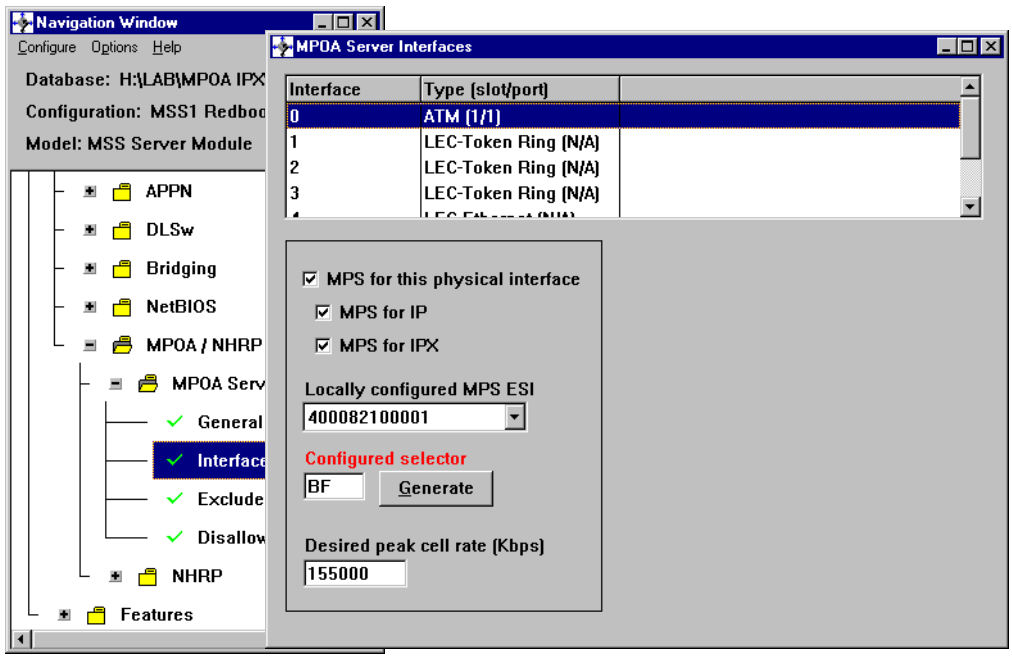


Figure 89. MPOA Advanced Setup for MSS_1

In case there is a need to configure an Exclude List or Disallowed Shortcut List, this can be done from the configuration program or from the command line. In *Talk 6*, use the commands

- `ADD EXCLUDE-LIST IPX` or
- `ADD DISALLOWED-ROUTER-TO-ROUTER-SHORTCUTS IPX`

Note: the word IPX is needed (or IP if configuring for this protocol) although the online help does not indicate it.

The Exclude List is used to identify routers that are on the routed path but do not support the NHRP Server function. The MPOA Server then responds to a Next Hop Resolution Request by providing the ATM address of this nexthop router instead of forwarding the Next Hop Resolution Request to it.

The Disallowed Rtr-Rtr Shortcut List should be used to prevent loops in some special cases. Operation of NHRP may result in establishing transit paths across NBMA networks between routers. However, establishing an NHRP shortcut across a boundary where information used in route selection is lost may result in a routing loop. Such situations include the loss of BGP path vector information, and the interworking of multiple routing protocols with dissimilar metrics. Under such circumstances, NHRP shortcuts between routers should be disallowed. This situation can be avoided if there are no “back door” paths between the entry and egress router outside the NBMA network.

MSS_2 was configured with the command line using the commands shown in Figure 90 on page 187.

```
MSS2 *TALK 6

MSS2 Config>PROTOCOL mpoa
Next Hop Resolution Protocol/Multi Protocol Over ATM user configuration
MSS2 MPOA config>MPS

MPOA Server user configuration
MSS2 MPS Configuration> ENABLE
Enable MPS IP for the box? [Yes]:

MPS will be enabled for IP on the box
Enable MPS IPX for the box? [Yes]:

MPS will be enabled for IPX on the box
MSS2 MPS Configuration> ADVANCED-CONFIG

MPOA Server advanced configuration
Interface number:  [0]?

MPS Physical Interface Configuration
MSS2 MPS Physical Interface Config> ESI
        [1] Burned-in-Address
        [2] 40.00.82.10.00.02
 ESI:  [2]? 2
MSS2 MPS Physical Interface Config> SELECTOR
MPS Selector (in hex) [2]? BF
MSS2 MPS Physical Interface Config> LIST

             MPS configuration on physical interface 0

  Net 0 status:   IP Enabled and  IPX Enabled
  ESI:                40.00.82.10.00.02
  Selector:                          BF
  Desired Peak Cell Rate (Kbps):  155000
  Line Speed (Mbps):                 155

MSS2 MPS Physical Interface Config> EXIT
MSS2 MPS Configuration>
```

Figure 90. MPOA for IPX Configuration in MSS_2 Using the Command Line

9.2.2 MPOA Client for IPX Configuration

To configure the MPOA Client in the MSS Client, the source-route bridge (SRB) function has to be enabled. The interfaces that will be served by the MPOA Client have to belong to the SRB.

To enable the MPOA Client in the MSS Client the following steps have to be followed:

1. Configure the ATM interface with a LANE Client to the appropriate ELAN and enable it for SRB also.
2. Configure SRB with the proper interfaces enabled.
3. Enable the MPOA Client for IPX at box level.
4. Configure global MPOA Client parameters.

The reader is assumed to be familiar with basic MSS Client configuration and in particular with the configuration of LECs and with concepts of configuring routing and bridging. For more information on this functions and how to configure them in

the MSS Client please see the manuals and redbooks referred to in Appendix D, “Related Publications” on page 221.

Steps 1 and 2 are common to any typical configuration for the MSS Client when it is used as a proxy to connect legacy token-rings to ATM.

The MPOA Client is enabled by default for both IP and for IPX. Moreover, if the MPOA Client is to be disabled permanently, then it is not enough to delete the MPOA client, as another one will automatically be created at the next reboot. It is necessary to disable the MPOA Client instead, and this might be needed when configuring the MSS Client for routing and using NHRP. Only one MPOA Client can exist per ATM interface, and since the MSS Client also only supports one ATM interface, only one MPOA Client can be configured in each MSS Client.

The default MPOA Client parameters usually work well, so the last step above is not normally required. In addition, the MPOA Client can be configured automatically using pre-defined values retrieved from the LECS. This option is also enabled by default, leading to an almost plug-and-play” environment for MPOA Clients.

As with the MPOA Server, MPOA Clients can be configured with the configuration program or with the command line. *Talk 6* and *Talk 5* can be used, with almost the same set of commands. Changes made in *Talk 6* can be activated dynamically in *Talk 5* using the RESET command.

Steps 1 and 2 above will not be covered in this book. For steps 3 and 4, Figure 91 on page 190 and Figure 92 on page 190 show how we configured one of our MSS Clients using the configuration program; Figure 93 on page 191 shows how we configured the other MSS Client using the command line. A brief description of the parameters involved follows. We recommend changing the ESI for the MPC because it makes debugging and problem tracing more simple.

- **Fragmentation mode:** Controls the manner that the ingress MPC handles IP packet fragmentation. If this parameter is set to *perform fragmentation*, normal fragmentation will take place. With any of the other two values, no fragmentation occurs. When this parameter is set to *maximize shortcut usage*, frames requiring fragmentation will be sent to the MPOA server, while smaller frames will be sent over the shortcut. A potential consequence of using *maximize shortcut usage* is that packets can get out of order.

  When this parameter is set to *maximize inorder usage*, usage of a particular shortcut will be suspended for the hold down time if a frame requiring fragmentation is received, causing all frames for the destination to be sent to the MPOA Server.

  Note that fragmentation applies only to IP since there is no IPX fragmentation. The default setting is to *perform fragmentation*.

- **LECS auto configuration parameters accepted:** This parameter indicates whether configuration parameters from the LECS will be accepted by the MPOA Client. The default value is *enabled*.

- **Shortcut setup frame parameters:** Controls the rate of traffic required before the client will initiate a shortcut to a given protocol destination.

  The client will initiate a shortcut when the number of frames specified by the value c*ount* is forwarded to the same protocol destination for a period of seconds specified by *time*.

- **LANE shortcuts:** This parameter determines whether LANE shortcuts are to be used for this client interface. LANE shortcuts are used by the MSS Client to establish shortcuts to non-MPC LANE Clients (only in MPOA for IP) and also between two MSS Clients (in both MPOA for IP and MPOA for IPX). LANE shortcuts take advantage of the hardware forwarding capabilities in the MSS Client when it is in the role of egress device.
- **Source MAC address origin:** This parameter specifies the source MAC address to be used in frames transmitted on LANE shortcut VCCs. Valid values are:
  - The burned-in MAC address in the client's ATM device
  - Locally-administered MAC
  - The MAC address provided in the MPOA resolution reply

  In general the default of using the ingress MPOA Client's burned-in MAC address should be used as it is appropriate for most networks and does not require any user configuration.

  One case where this may lead to a problem is when the burned-in MAC address is also being used by a local LEC that is a member of the same ELAN as the LEC to which the shortcut is established. The problem that may occur is that the destination LEC can become confused about which VCC the burned-in MAC address is associated with if the destination LEC learns MAC address-to-VCC bindings from received frames. In networks where this problem arises, a locally-administered MAC Address may be configured for use as the source MAC address.

  It is still possible, though unlikely, that use of either the burned-in MAC address or a locally-administered MAC address could confuse some endstation protocol stack implementations, since the source MAC address will not match that of the router that the endstation uses as a gateway to transmit packets to the associated protocol address. In the case of IP, for this to happen, the endstation would have to learn router MAC addresses from unicast IP frames. This is not normal behaviour since IP-to-MAC address mappings are normally learned from ARP packets. If such a situation were to be encountered, the MPC can be configured to use the MAC address provided in the MPOA Resolution Reply as the source MAC address; this is the MAC address of the egress router.

  Using the last hop router's MAC address as the source MAC address solves this problem of endstation protocol stack confusion but introduces another potential problem. It may confuse LECs that learn MAC address-to-VCC bindings from received frames (since the LEC may decide that frames destined for the router should be sent over the shortcut VCC), and therefore should not be used with LECs that perform this type of learning (note that the LEC in the IBM 8281 ATM-LAN Bridge is one example of a LEC that performs this type of learning).

  If none of these options is suitable for a given network, shortcuts to all LANE devices may be disabled altogether. shortcuts to particular protocol destinations may also be disabled using the MSS Server's NHRP Exclude List configuration option.
- Other parameters should be left as the default in most networks.

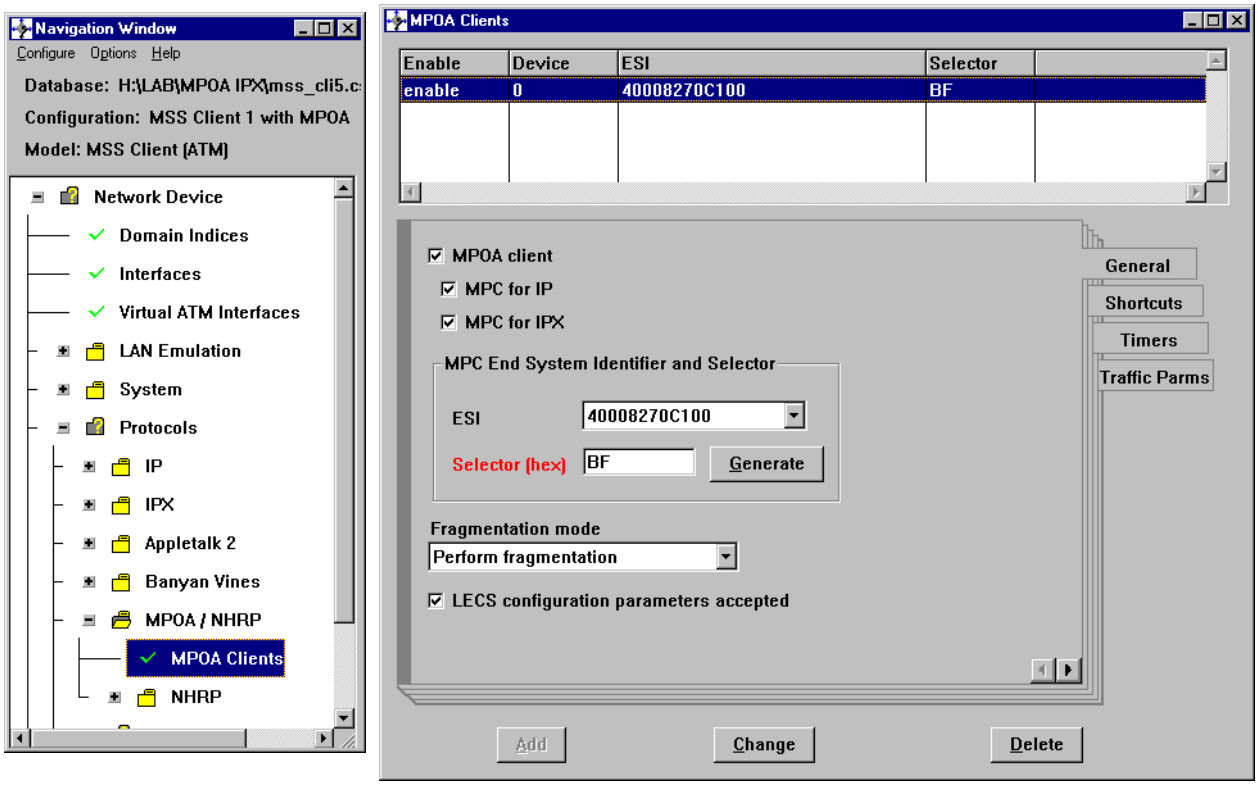


Figure 91. MPOA Client General Configuration

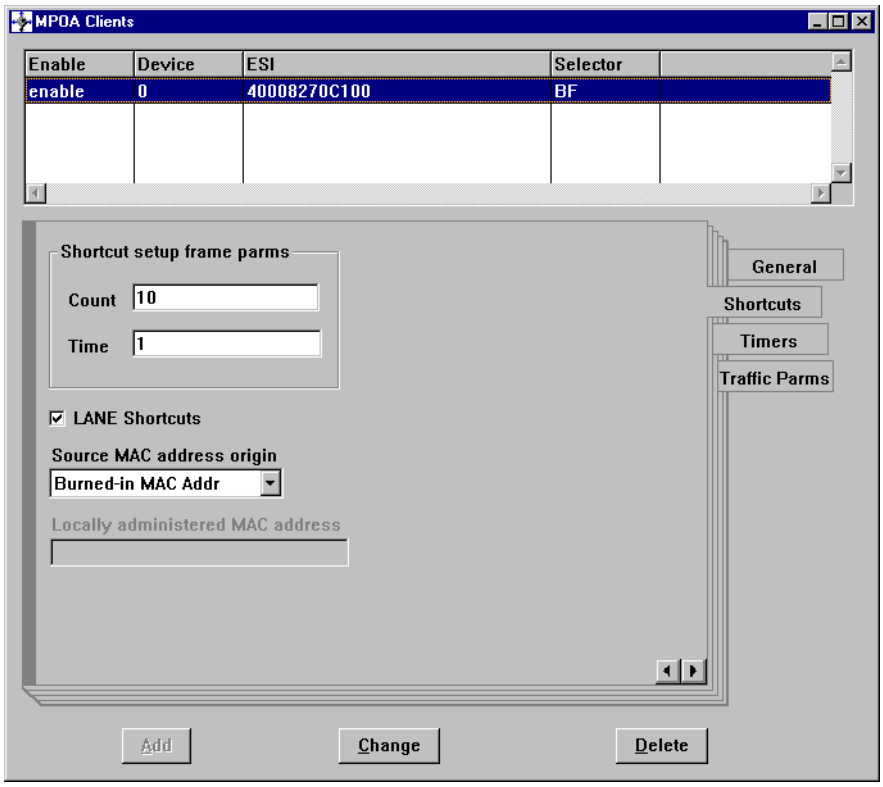


Figure 92. MPC Shortcuts Configuration

```
MSSC_2 *TALK 6

MSSC_2 Config>PROTOCOL mpoa
Next Hop Resolution Protocol/Multi Protocol Over ATM user configuration
MSSC_2 MPOA config>MPC

MPOA Client user configuration
==============================
MSSC_2 MPC >ADD

     MPC added on interface 0
MSSC_2 MPC >CONFIG
MSSC_2
MPC Configuration> ENABLE

    MPC is already ENABLED
MSSC_2
MPC Configuration> SET SELECTOR
Selector Byte (in hex) [2]? BF
MSSC_2
MPC Configuration> SET ESI
        [1] Burned in ESI
        [2] 40.00.82.70.C2.00
 ESI:  [1]? 2

MSSC_2
MPC Configuration> SET IPX-PROTOCOL
Enable IPX (Yes/No):? [Yes]: y
MSSC_2
MPC Configuration> LIST

   MPC Configuration
   -----------------
   STATUS: ENABLED
   Shortcut Setup Frame Count:       10    (frames)
   Shortcut Setup Frame Time:        1     (sec)
   Initial Retry Time:               5     (sec)
   Maximum Retry Time:               40    (sec)
   Hold Down Time:                   160   (sec)
   VCC Timeout Period:               20    (min)
   Accept Config From LECS:          Yes
   Fragmentation Mode:               Perform Fragmentation

   Interface:                        0
   ESI:                              40.00.82.70.C2.00
   Selector:                         0xBF
   Desired PCR:                      155000 (kbps)
   Maximum Reserved Bandwidth:       155000 (kbps)
   Line Rate:                        155 (Mbps)
   Enable LANE Shortcuts:            TRUE
   Source MAC Address for Shortcuts: Burned In

   IP-Protocol: ENABLED
   IPX-Protocol: ENABLED

MSSC_2
MPC Configuration>
```

Figure 93. MPOA Client for IPX Configuration Using Command Line

9.2.3 Monitoring MPOA for IPX

As you can see from Figure 84 on page 180, there are two MPOA Servers and three ELANs in our network. The intermediate ELAN is an Ethernet ELAN, which uses different IPX encapsulation from the token-ring networks. Different IPX encapsulation is also used in the two token-ring networks, to show that encapsulation does not have to be the same all the way through the network.

Our IPX server was configured with encapsulation type *Token-Ring MSB* and network number 1005. The internal network number was 1000.

Some ordinary ELAN clients that are not MPOA Clients are used in our network. The MPOA Server for IPX does not support shortcuts to these stations.

MPOA was designed from the start to implement shortcuts between different LAN media types, such as in the case of a shortcut between an MSS Client (token-ring) and a 8371 Multilayer Ethernet Switch. See 3.3, “MPOA across Mixed-Media Networks” on page 58 for an example of MPOA between Ethernet and token-ring networks.

Many commands can be used to verify that MPOA is working properly and to check how shortcuts are being established. In this section we will explain some of them.

9.2.3.1 Checking the Status of MPOA in the MSS Server

To display the active configuration of MPOA at the box level, use the command-line interface and use the following sequence of commands:

```
talk 5
protocol mpoa; mps; display
```

One of the advantages of MPOA is that it discovers all neighbour MPOA Servers and MPOA Clients in the network. To display the information that the MSS Server has discovered automatically, use the `DISCOVERY` command.

To display the status of the control VCCs use the `CONTROL-VCCs` command. Control VCCs are used between MPOA entities to exchange MPOA control information.

Figure 94 on page 193 shows an example of both of these last two commands executed in MSS_1.

```
MSS1 *TALK 5

MSS1 +PROTOCOL mpoa
MSS1 MPOA>MPS

MPOA Server Console
MSS1 MPS >DISCOVERY
                          DISCOVERY TABLE
 Net  Type  Age MAC Addr/RD            ATM Address
 ---- ---- ---- ------------ ----------------------------------------
 1     MPC 103  40008270C102 39999999999999000099990101400082700100BF
 2     MPS 224  400082101802 39999999999999000099990102400082100000200
 2     MPC 224  40008270C202 39999999999999000099990102400082700200BF
 6     MPS 43   400082101902 39999999999999000099990102400082100000200
 1     MPC 284          0041 39999999999999000099990101400082700100BF
 1     MPC 43           0051 39999999999999000099990101400082700100BF
 2     MPC 1125         0062 39999999999999000099990102400082700200BF
MSS1 MPS >CONTROL-VCCs
                    MPS Control VCCs
VPI  VCI  Net RefCnt Remote ATM Address
--- ----- --- ------ ----------------------------------------
  0   123   0 2      39999999999999000099990102400082100000200
  0   161   0 8      39999999999999000099990101400082700100BF
  0   154   0 2      39999999999999000099990102400082700200BF
MSS1 MPS >
```

Figure 94. Checking the Status of MPOA in the MPS

9.2.3.2 Checking the Status of MPOA in the MSS Client

Similar information can be obtained in the MSS Client.

To display the actual configuration of MPOA use the following sequence of commands:

```
talk 5
protocol mpoa; mpc; configure; list
```

To check the neighbours use the command `LIST` in the NEIGHBOR-MPSs menu.

Figure 95 on page 194 shows an example of these commands executed in MSSC_2.

```
MSSC_2 +PROTOCOL mpoa
MSSC_2 MPOA>MPC

MPOA Client Console
==================
MSSC_2 MPC >CONFIGURE

MPOA Client Dynamic Configuration
=============================
MSSC_2 MPC CONFIGURE>LIST
          MPOA Client Configuration:
          ======================

   Status:                           ENABLED
   Shortcut Setup Frame Count:       10
   Shortcut Setup Frame Time:        1      (sec)
   Initial Retry Time:               5      (sec)
   Maximum Retry Time:               40     (sec)
   Hold Down Time:                   160    (sec)
   VCC Timeout Period:               20     (min)
   Accept Config From LECS:          Yes
   Fragmentation Mode:               Perform Fragmentation
   Interface:                        0
   ESI:                              40.00.82.70.C2.00
   Selector:                         BF
   Desired PCR:                      155000   (kbps)
   Maximum Reserved Bandwidth:       155000   (kbps)
   Line Rate:                        155      (Mbps)
   Enable LANE Shortcuts:            TRUE
   Source MAC Address for Shortcuts: Burned In

          Packet Trace Filtering Parameters:
          ==============================

   ATM Address Pkt Trace Filter Value
    00000000000000000000000000000000000000000000
   ATM Address Pkt Trace Filter Mask
    00000000000000000000000000000000000000000000
   LAN Pkt Trace Filter Value
    000000000000
   LAN Pkt Trace Filter Mask
    000000000000

MSSC_2 MPC CONFIGURE>EXIT
MSSC_2 MPC >NEIGHBOR-MPSs

MPOA Client MPS Console
=======================
MSSC_2 MPC MPS>>LIST
          List of Neighbor MPSs for MPOA Client (interface 0):
          ============================================
 1) Control ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00

     1 MAC Address(es) Learnt For This MPS:

      1) MAC Addr: x40.00.82.10.18.01   Associated LEC Intf #:  1
 2) Control ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00

     1 MAC Address(es) Learnt For This MPS:

      1) MAC Addr: x40.00.82.10.18.02   Associated LEC Intf #:  1
```

Figure 95. Checking the MPOA Status on the MSS Client

9.2.3.3 Displaying the Established Shortcuts on MPCs

To display the information about the active shortcuts, use the command `LIST` in the VCCs menu of the MPC. This will show a list of all VCCs that are related to this MPC (except control VCCs), and the state of each of them.

Having displayed all the VCCs, to then view details on a specific VCC use the command `LIST-VCC`.

```
MSSC_2 MPC >VCCs

MPOA Client VCC Console
=======================
MSSC_2 MPC VCC>LIST                                                    1
         SVCs For MPC On ATM Interface 0 (total    3):
         ========================================
   1) VPI/VCI 0/615    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00
   2) VPI/VCI 0/616    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00
   3) VPI/VCI 0/621    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF

MSSC_2 MPC VCC>LIST-VCC
VPI, Range 0..255 [0]?
VCI, Range 0..65535 [0]? 621                                           2

  VPI/VCI: 0/621     State: OPERATIONAL    Calling Party: FALSE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:4544/4544
  Remote ATM Addr: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: TR-LANE
  H/W Path Valid: TRUE    Ref. Frame Cnt: 9639
  Frames Tx/Rx: 813/8829     Bytes Tx/Rx: ?/?


      (Direct) Shortcut Routes Using This VCC:
      ----------------------------------------                         3
     1) Network Number (in hex): 1000    State: RESOLVED


MSSC_2 MPC VCC>
```

Figure 96. Displaying Shortcut VCCs on MSSC_2

Figure 96 is an example of these commands on MSSC_2. The information shown is:

1 List of all VCCs to this MPC

2 Detail of VCC 0.621. The ATM address of the other end of this SVC is the other MSS Client. This shortcut is the one that connects PC105 to the IPX server.

The most relevant parameters shown have the following meaning:

- **State**: Status of the VCC. This display will include VCCs which are not fully operational.
- **Calling Party:** Is TRUE when this MPC is the one that originated the SVC. In this case, the SVC was originated by MSSC_1, so this field is FALSE.
- **Fwd/Bak SDU:** Indicates the MTU size that is being used in both directions

- **Encaps. Type:** Shows the encapsulation of this VCC. It can be 1483 or LANE. Remember that MSS Clients will always attempt to use LANE VCCs to other MSS Clients for performance reasons.
- **H/W Path Valid**: If TRUE means that the MSS Client is actually making use of hardware forwarding capabilities for *received* frames. Frames received over this VCC are switched in hardware to the token-ring ports. This applies, as in this case, when the MSS Client is acting as the egress MPOA Client for this shortcut and LANE encapsulation is being used.
- **Frames Tx/Rx:** Shows the number of frames transmitted and received over this VCC. Here there are frames flowing in both directions. Most shortcut VCCs will be used in both directions, although sometimes parallel activity by two MPOA Clients can result in two separate VCCs being set up, in which case traffic will switch to using one VCC after a short time and the redundant VCC will eventually age out and be dropped.

3 This lists all the shortcuts that are actually flowing through this SVC. It confirms that this SVC is used to connect PC105 to the IPX server. The entry shown for the shortcut points to network 1000 which is the internal network of the IPX server.

Figure 97 shows the result of the same displays at the other end of the shortcut, at MSSC_1.

```
MSSC_1 MPC >VCCs

MPOA Client VCC Console
=======================
MSSC_1 MPC VCC>LIST
          SVCs For MPC On ATM Interface 0 (total    3):
          ========================================
   1) VPI/VCI 0/150    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00
   2) VPI/VCI 0/153    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.42.00.82.72.00.00.80
   3) VPI/VCI 0/155    State: OPERATIONAL
        Remote ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF

MSSC_1 MPC VCC>LIST-VCC 0 155

  VPI/VCI: 0/155     State: OPERATIONAL    Calling Party: TRUE
  Hold Down Cause:  N/A   Cause Code: N/A   Fwd/Bak SDU:4544/4544
  Remote ATM Addr: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
  Conn Type: P2P     VCC Type: B. EFFORT     Encaps. Type: TR-LANE
  H/W Path Valid: TRUE    Ref. Frame Cnt: 19594
  Frames Tx/Rx: 9949/9645     Bytes Tx/Rx: ?/?


      (Direct) Shortcut Routes Using This VCC:
      ----------------------------------------
     1) Net/Host (in hex): 2005/40.00.01.05.00.0A    State: RESOLVED


MSSC_1 MPC VCC>
```

Figure 97. Displaying Shortcut VCCs on MSSC_1

9.2.3.4 Displaying the Imposition Cache Information MPSs

It is possible to view the cache information that an MPOA Server has saved for a particular shortcut. This will enable us to see the DLL header that the egress MPC will put on the outgoing frame (in the case of RFC1483 shortcuts; the DLL header is supplied by the ingress MPC for LANE shortcuts). This header should correspond to the one that the frame would have had if it had gone through the normal routed path.

Use the command `IMPOSITION-CACHE LIST` to view all the entries in the cache. Then use the command `IMPOSITION-CACHE ENTRY` to view details about one entry, as shown in Figure 98. This figure shows the result of this command in MSS_2. The cache entry corresponds to the station named PC105 in the drawing of Figure 84 on page 180. *I-MPS* refers to the ingress MPOA Server, which is MSS_1 in this case because it is the one that received the Resolution Request from the MPOA Client initiating the shortcut.

```
MSS2 *TALK 5

MSS2 +PROTOCOL mpoa
MSS2 MPOA>MPS

MPOA Server Console
MSS2 MPS >IMPOSITION-CACHE LIST

  Total Cache Entries = 1
  MPOA Imposition Cache Entries
  ============================
 CacId  Destination Address     NextHop Address    State Htime Prot
 ----- -------------------- -------------------- ----- ----- ----
 1         2005/40000105000A     2005/40000105000A Act   2000  IPX
MSS2 MPS >IMPOSITION-CACHE ENTRY
CacheId [1]? 1
 CacheId:     1
 State:       Active
 Elan-id:     8
 I-MPS Addr:  3005/400082101901
 Destination: 2005/40000105000A
 NextHop:     2005/40000105000A
 HoldingTime: 1994 seconds
 MTU size:    4376
 Prefix:      0x50
 Elan-type:   Token Ring
 DLH Length:  28
 DLHeader:    004040000105000AC0008210180206B000620020AAAA030000008137
 I-MPC data ATM: 39999999999999000099990101400008270C100BF
 E-MPC data ATM: 39999999999999000099990102400008270C200BF
MSS2 MPS >
```

Figure 98. Displaying Imposition Cached Information on MPS

9.2.3.5 Displaying the Cached Information in MPCs

The next four figures show both the ingress and the egress cache entries for both MPOA Clients, MSSC_1 and MSSC_2.

Some points arise from all four of these displays:

- All the shortcuts are using LANE encapsulation. This is apparently contradicted by the egress LIST-IPX display, in which only entries for MPOA-TAG and Native 1483 are shown.

- The ingress cache entries show the encapsulation types of TR-802.2-IPX-LANE and TR-SNAP-IPX-LANE. The difference is accounted for by the fact that the token-ring IPX encapsulation format is different between the two real token-ring segments in our network.
- The ingress cache entries include the DLL header to be used by the egress MPOA Client. At first glance these entries do not appear to match the corresponding entries in the other MPOA Client's cache. This apparent discrepancy can be resolved when it is realised that the DLL Header shown in the egress cache entries is a combination of the LANE Encaps. Hdr and the Encaps. Type displays (see Figure 99 on page 199) of the ingress cache entries shown, because:
  - Token-ring 802.2 IPX headers end with *e0e003*
  - Token-ring SNAP IPX headers end with *aaaa030000008137*

  These values agree with the encapsulation formats shown in Figure 83 on page 178.

```
MSSC_2 MPC INGRESS>LIST-IPX
          IPX-Ingress Cache For MPC on ATM Interface 0
          ===================================


          Ingress Cache Entries for Direct Host Routes:
          -----------------------------------------

          Ingress Cache Entries for Direct Network Routes:
          -----------------------------------------------
   1) Network Number (in hex): 1000     State: RESOLVED

          Ingress Cache Entries for Derived Host Routes:
          ---------------------------------------------
   1) Net/Host (in hex): 1000/00.00.00.00.00.01     State: RESOLVED
Derived From: 1000

MSSC_2 MPC INGRESS>LIST-ENTRIES-IPX
Destination Network Number (in 8-digit hex) (1 - FFFFFFFE) [0]? 1000
    Destination Node Number (in hex) (0x000000000000 for network destination):
[00.00.00.00.00.00]?
      Host Route Entries matching 1000/000000000000
      --------------------------


          Direct Host Routes :


          Derived Host Routes :

1) Network Number (in hex): 1000  State: RESOLVED
      Hold Down Cause: N/A   CIE Code: x0
      Dest ATM: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF
      Frames Sent To MPS: 0  Frames Sent Over Shortcut: 9646
      Remaining Age (mins:secs): 17:6      Last Request ID: x8
      Destn MTU: 4381       Encaps. Type: TR-802.2-IPX-LANE
      LANE Encaps. Hdr: x00000040420080880050c0008210170106b000510010
      Tag Value: N/A
      Shortcut VCC (VPI/VCI):    0/ 621   Local Shortcut ?: FALSE
      MPS: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00

      Network Route Entries matching 1000
      -----------------------------

   1) Net/Host (in hex) : 1000/000000000001  State: RESOLVED
      Hold Down Cause: N/A   CIE Code: x0
      Destn: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF
      Frames Sent To MPS: 0  Frames Sent Over Shortcut: 9646
      Remaining Age (mins:secs): 17:6
      Last Request ID: x8
      Destn MTU: 4381       Encaps. Type: TR-802.2-IPX-LANE
      LANE Encaps. Hdr: x00000040420080880050c0008210170106b000510010
      Tag Value: N/A
      Shortcut VCC (VPI/VCI): 0/621  Local Shortcut ? FALSE
      MPS: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00

MSSC_2 MPC INGRESS>
```

Figure 99. Ingress Cache on MSSC_2

```
MSSC_2 MPC EGRESS>LIST-IPX
          Egress Cache For MPC on ATM Interface 0
          ======================================

      Egress Cache Entries w/ MPOA-Tag Encapsulation:
      -----------------------------------------------



      Egress Cache Entries w/ Native 1483 Encapsulation (Host Routes):
      ---------------------------------------------------------------

   1) Net/Host (in hex): 2005/40.00.01.05.00.0A     State:     ACTIVE


      Egress Cache Entries w/ Native 1483 Encapsulation (Netwk Routes):
      ---------------------------------------------------------------



MSSC_2 MPC EGRESS>LIST-ENTRIES-IPX
Destination Network Number (in 8-digit hex) (1 - FFFFFFFE) [0]? 2005
    Destination Node Number (in hex) (0x000000000000 for network destination):
[00.00.00.00.00.00]?

      Egress Cache Entries matching 2005/000000000000 :

   1) IPX Net /Host Num: 2005/40000105000a   Entry Type: 1483 (HOST, DIRECT)
    LEC #:  1    Cache ID: x2   State: ACTIVE
    MPS: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.10.00.02.00
    Source: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.70.C1.00.BF
    Remaining Age (mins:secs): 26:51
    Recvd Octets: N/A
    Recvd Frames Forwarded: N/A
    Recvd Frames Discarded: N/A
    Tag Value: N/A       Local Shortcut: FALSE
    DLL Header: x004040000105000ac0008210180206b000620020aaaa030000008137
    LANE Extensions in last Imposition reply:  Formats 7, 11, 13, 17
```

Figure 100. Egress Cache on MSS_2

```
MSSC_1 MPC INGRESS>LIST-IPX
          IPX-Ingress Cache For MPC on ATM Interface 0
          ===========================================


          Ingress Cache Entries for Direct Host Routes:
          ---------------------------------------------
   1) Net/Host (in hex): 2005/40.00.01.05.00.0A  State: RESOLVED

          Ingress Cache Entries for Direct Network Routes:
          ------------------------------------------------

          Ingress Cache Entries for Derived Host Routes:
          ----------------------------------------------

MSSC_1 MPC INGRESS>LIST-ENTRIES-IPX
Destination Network Number (in 8-digit hex) (1 - FFFFFFFE) [0]? 2005
    Destination Node Number (in hex) (0x000000000000 for network destination):
[00.00.00.00.00.00]?
      Host Route Entries matching 2005/000000000000
      -----------------------------


          Direct Host Routes :

1) Net/Host (in hex): 2005/40000105000a  State: RESOLVED
      Hold Down Cause: N/A   CIE Code: x0
      Dest ATM: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
      Frames Sent To MPS: 50  Frames Sent Over Shortcut: 1120
      Remaining Age (mins:secs): 15:21      Last Request ID: x4C
      Destn MTU: 4376       Encaps. Type: TR-SNAP-IPX-LANE
      LANE Encaps. Hdr: x00000040400001050000ac0008210180206b000620020
      Tag Value: N/A
      Shortcut VCC (VPI/VCI):    0/ 155   Local Shortcut ?: FALSE
      MPS: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00


          Derived Host Routes :

      Network Route Entries matching 2005
      -------------------------------

None found!

MSSC_1 MPC INGRESS>
```

Figure 101. Ingress Cache on MSSC_1

```
MSSC_1 MPC >EGRESS-CACHE

MPOA Client Egress Cache Console
================================
MSSC_1 MPC EGRESS>LIST-IPX
          Egress Cache For MPC on ATM Interface 0
          ======================================

      Egress Cache Entries w/ MPOA-Tag Encapsulation:
      -----------------------------------------------



      Egress Cache Entries w/ Native 1483 Encapsulation (Host Routes):
      ---------------------------------------------------------------



      Egress Cache Entries w/ Native 1483 Encapsulation (Netwk Routes):
      ----------------------------------------------------------------

   1) Network Number (in hex): 1000     State:     ACTIVE


MSSC_1 MPC EGRESS>LIST-ENTRIES-IPX
Destination Network Number (in 8-digit hex) (1 - FFFFFFFE) [0]? 1000
    Destination Node Number (in hex) (0x000000000000 for network destination):
[00.00.00.00.00.00]?

      Egress Cache Entries matching 1000/000000000000 :

   1) IPX Net /Host Num: 0   Entry Type: 1483 (NETWK, DIRECT)
    LEC #:  1    Cache ID: x1   State: ACTIVE
    MPS: 39.99.99.99.99.99.99.00.00.99.99.01.01.40.00.82.10.00.01.00
    Source: 39.99.99.99.99.99.99.00.00.99.99.01.02.40.00.82.70.C2.00.BF
    Remaining Age (mins:secs): 33:52
    Recvd Octets: N/A
    Recvd Frames Forwarded: N/A
    Recvd Frames Discarded: N/A
    Tag Value: N/A       Local Shortcut: FALSE
    DLL Header: x00404200808800050c00082101701060b000510010e0e003
    LANE Extensions in last Imposition reply:  Formats 7, 11, 13, 17


MSSC_1 MPC EGRESS>
```

Figure 102. Egress Cache on MSSC_1

9.3 MPOA Server MIB

The availability of the MPOA Server MIB means that network management applications can now monitor the MPOA Server through SNMP.

The first MIB release on the MSS Server will support the *mpoaMpsBasicCompliance* MIB; not all the defined MPOA MIB objects in the tables will be supported in the MSS Server.

This feature can be used with existing MSS configurations that have MPOA server support. It can be used with other IBM or non-IBM products which want access to MPOA Server information using SNMP.

9.4 MPOA Client MIB

In a similar manner as the MPOA Server MIB, this first release of MIB support in the MPOA Client will support the *mpoaMpcBasicCompliance* MIB compliance. Not all MPOA MIB Client objects in the tables will be supported in the MPOA Client running on the MSS Client.

9.5 MPOA Enhancements Conclusion

Although we have spent a chapter documenting how to configure and monitor MPOA for IPX, much of what has been shown is identical to the configuration and monitoring of MPOA for IP: once IPX routing has been configured in the MSS Servers in a network there remains little to do to enable MPOA for IPX other than to switch it on in the MPOA Servers and in the MPOA Clients. Although TCP/IP remains the “protocol of choice” for most networks, many existing networks can benefit from the ability of ATM networks to perform Layer 3 Switching for IPX.

Chapter 10. MSS V2.2 Miscellaneous Enhancements

This chapter contains additional information on the other new features and enhancements included in V2.2. See Chapter 5, “MSS V2.2 New Features and Enhancements Summary” on page 91 for a complete list of all the new features and enhancements in this release of code; this chapter provides additional explanation where necessary for any of the items listed but not covered elsewhere in this book.

10.1 OSPF Version 2 (RFC 2178)

RFC 2178 documents OSPF Version 2. Appendix G of this RFC documents the differences between OSPF Version 2 and OSPF Version 1 (as originally documented in RFC 1583) as:

1. Enhancements to OSPF authentication
2. Addition of point-to-multipoint interface
3. Support for overlapping area ranges
4. A modification to the flooding algorithm
5. Introduction of the MinLSArrival constant
6. Optionally advertising point-to-point links as subnets
7. Advertising same external route from multiple areas
8. Retransmission of initial database description packets
9. Detecting interface MTU mismatches
10. Deleting the TOS routing option

10.2 TOS

The Common Code family of routers implements support for both setting and observing the IPv4 Precedence field in the IPv4 header. Support in MSS is limited to the former - that of setting the bits in appropriate circumstances - because the act of observing and acting on the Precedence field is performed by the Bandwidth Reservation System (BRS), which does not run in MSS. Any reference to BRS below, therefore, refers to the ability of other routers in the network to act on the IPv4 Precedence field settings made by MSS.

10.3 IP MTU by Interface

Previous releases of code did not allow explicit configuration of the IP maximum transmission unit (MTU) size. MSS would calculate the MTU size on an interface by taking the (configurable) maximum frame size and subtracting the size of the Layer 2 header. This code release allows specification of a lower IP MTU size than derived through this calculation. The ability to specify a different MTU size than the default calculated MTU size is especially important when using OSPF, because OSPF advertisements include the MTU size in the advertisements themselves; adjacent routers will not participate in OSPF topology exchanges if they determine that a neighbour’s MTU size is too large.

10.4 APPN Configuration TG Number

APPN code in MSS now allows the option of configuring the transmission group (TG) number on an APPN link. Normal APPN operation allows for the negotiation of TG numbers, and these numbers are allocated starting at 21 and proceeding upward, as can be seen in Figure 103 on page 206, which shows an ATM link station that has been created dynamically in response to a connection request received over an Ethernet ELAN.

```
MSS2 APPN >LIST LINK_INFORMATION
    Name   Port Name  Intf       Adj CP Name  Type       HPR      State
=============================================================================
    @@16      E00004     4  USIBMRA.LAB2210A    NN    ACTIVE     ACT_LS
MSS2 APPN >LIST LINK_INFORMATION @@16

Link Station Information
========================================
        ls_name = @@16
        type = DYNAMIC
        act_at_startup = FALSE
        auto_act_supported = FALSE
        pan uplink = FALSE
        adjacent node subnet affiliation = NATIVE
        subnet visit count = 0
        remote mac_addr (non-canonical)= 4000440855FF
        remote mac_addr (canonical)= 02002210AAFF
        remote sap_value = 04
        hpr_sap_value = C8
        real_adj_cp_name = USIBMRA.LAB2210A
        node_id = 00000000
        cp_cp_sessions_supported = TRUE
        hpr_supp = TRUE
        hpr link = TRUE
        link station state = ACT_LS
        direction = INBOUND
        actual_max_send_btu_size = 1289
        partner_node_type (actual) = NN
        partner_node_type (defined) = LEARN
        tg_isr_type = INTERMEDIATE_ROUTING_TG
        tg_num (defined) = 0
        tg_num (actual) = 21
```

Figure 103. APPN Link Station Information

The MSS manual states, “When parallel TGs over ATM are configured, the adjacent node name and link station must be defined in both nodes for each link station”. This requirement is imposed by the architecture defined for the implementation of APPN/HPR over ATM.

10.5 APPN APING

APING is the APPN equivalent to the TCP/IP `PING` command and can be used to test connectivity to other ATM devices:

```
MSS2 APPN >APING usibmra.lab2210a
Allocate duration: 0 msec
Iteration  Duration  Data Sent  Data Rate
  number    (msec)    (bytes)     (Kb/s)     LU name
-----------------------------------------------------
     0       110        100          7       USIBMRA.LAB2210A
-----------------------------------------------------
Avg.         110        100          7
MSS2 APPN >APING ?
(End of command.  Confirm with Carriage Return)
MSS2 APPN >APING ?
 usage:
 aping  [flags] fq_lu_name
   -m  mode name                            ( default:  #INTER )
   -t  TP name                              ( default:  APING  )
   -i  count of sends & receives to issue   ( default:  1      )
   -x  count of conversations to run        ( default:  1      )
   -y  count of TPs to run                  ( default:  1      )
   -s  size of packet                       ( default:  100    )
   -q  quiet                                ( accepts no value )
   -b  background Display goes to Talk 2    ( accepts no value )
```

Figure 104. APING in Action

Part 3. Appendixes

Appendix A. Common Pitfalls with MSS Products

This appendix lists some of the common mistakes made when configuring the MSS Server and both types of MSS Family Client. For more information on any subject, please refer to the documentation provided on the CD-ROM and available at the following Web site:

`http://www.networking.ibm.com/nes/neshome`

1. MSS/ATM switch mismatch

   Take care to ensure that the UNI version settings of the MSS and the ATM switch are the same. This is often overlooked if the cable is moved to a new switch port after initial setup.

2. APPN TG number mismatch

   In MSS Release 2.2 the TG for CP-CP sessions is configurable. Ensure the same value is set at both ends of the session if it is configured (in most cases, this is unnecessary and the actual TG number will be determined by negotiation between the two control points).

3. IP access filters

   If IP access filters are used, remember to configure an *include* filter to allow all packets that are not excluded.

4. IP MTU size

   IP MTU size is configurable. Mismatched MTU sizes can cause problems, especially with OSPF.

5. IPX broadcast circuits

   IPX is no longer enabled on an interface basis; you must now add a broadcast circuit.

6. IPX Encapsulation with MPOA

   Ensure that the same IPX encapsulation type is used throughout the network.

7. BUS filters

   Be judicious with the use of BUS filters to avoid possible performance impact.

8. LES/BUS Enhancements and Mode Type

   Many of the LES/BUS enhancements (for example, broadcast manager) require the LES to be in *system* mode and not in VCC splice or adapter mode

9. LES security

   If you are having problems with LE Clients not joining ELANS, check that LECS Security is configured correctly.

10. 1483 bridge ports

    Remember to set the MAC address if the lowest bridge port configured is a 1483 port, otherwise the 1483 port will not function correctly.

11. Both NHRP and MPOA are enabled by default.

    When configuring NHRP, MPOA should be disabled and vice versa. To disable MPOA function first add an MPOA Client and then disable it.

12. Configuring bridging and a routed protocol on the same interface in the MSS Family Client when the subnet of the routed protocol spans across the bridge.

By configuring the routed protocol and SRB on the same interface, the MSS Family Client will filter the protocol traffic, and prevent it from being SR bridged. If the subnet of the routed protocol spans the SR bridge, communication within the subnet will not be reliable. This can be resolved by having two network interfaces in a domain or ELAN that is to be SR bridged. One interface should be configured for SRB (without the routed protocol) and the other interface should be configured for the routed protocol.

13. Configuring the LAN Switch IP address to the same IP address of the MSS Family Client.

    Although the MSS Family Client and the LAN switch functions are closely integrated, they both have separate IP interfaces in the network. If this error is made, one or both interfaces might not be able to communicate with the network.

14. Configuring SRB only through LAN switch configuration options without configuring the MSS Family Client.

    To use the MSS Family Client's SRB function, it must be explicitly configured. When the MSS Family Client's SRB function becomes active, its configuration will take precedence over any previous LAN switch SRB configuration.

15. Using the Ctrl-B key to exit a MSS Family Client console when in a Telnet session.

    A user can access the MSS Family Client's console through the LAN switch console. If the user is accessing the LAN switch through the communication port located on the LAN switch, the Ctrl-B key sequence, or the Return command, can be used to exit the MSS Family Client's console and return to the LAN switch console. If a user Telnets to the LAN switch, the Ctrl-B key sequence will end the Telnet session. When using Telnet to the LAN switch, if the user wishes to exit the MSS Family Client's console, without terminating the telnet session, the Return command should be used from the * prompt of the MSS Family Client.

16. The MSS Family Client is configured for SRB in a LAN switch with an ATM UFC not at the proper HW level.

    If this is the case, the MSS Family Client console will show the SRB function to be operational, but there will be no SRB options from the LAN switch console.

17. Configuring IP-Host in the MSS Family Client when IP is already configured.

    This problem will result in the IP-Host being configured, but the user will not be able to get into the HST sub-menu of the *Talk 5* console. This can occur even if there is no IP address configured. The solution is to enter the `clear ip` command from the *Talk 6* menu, write the configuration, and then reset. This will remove IP from the configuration and allow the IP-Host support to work.

18. Configuring SRB on two interfaces in the same domain.

    This results in one of the two interfaces being disabled. Because a domain can represent one and only one ring number, having two interfaces bridged on one domain is incorrect.

19. Configuring two or more LAN Emulation Clients (LECs) on the same MSS Client to the same Emulated LAN (ELAN) without specifying a locally administered MAC address on one of them.

The default MAC address of the LEC is the burned-in address of the ATM interface of the MSS Client. If two LECs attempt to join an ELAN with the same MAC address, the ELAN will reject the registration. The solution is to assign a locally administered MAC address on all but the first LEC.

20. Attempting to access the MSS Family Client console by Telnetting to the LAN switch, while a console to the MSS Client is active either from the LAN switch COM port or other Telnet session.

    Only one session is permitted through the LAN switch to the MSS Family Client at a time.

21. Attempting to place a port in a domain that is used for bridging to an MSS Client LEC.

    The MSS Client supports SRB to LECs. It does *not* support transparent bridging. The MSS Client obtains a domain for each bridged LEC. Domains will be reserved starting from the maximum domain index (15) to the lowest domain index (0). Once these domains have been allocated, they cannot be used by other ports. This does not apply to LECs that are not being bridged.

    Example: The MSS Client is configured for three bridged LECs, two routed LECs, and two token-ring interfaces on domain 0, one bridged, one routed. The three bridged LECs will reserve domain indexes 15, 14, and 13. The token-ring interfaces will be a member of domain 0, which can have other ports as members. The routed LECs are not associated with any domains.

22. Attempting to configure more than one physical port in a domain that is to be managed by LNM.

    Although the bridging function will continue to work, LNM will not be able to manage the ring. If a domain is currently being managed by LNM and the domain configuration is changed, then the MSS Family Client should be reset. Failure to do so can cause LNM information to be unreliable.

23. Attempting to use the MSS Server Configuration Program to configure an MSS Family Client.

    The MSS Family Client must be configured using its version of the IBM MSS Client Configuration Program that is included on the CD-ROM or obtained from the Web. It is not compatible with the MSS Server version of the configuration program.

24. Attempting to use SNMP to manage the LAN switch and the MSS Family Client without enabling SNMP on both the LAN switch and the MSS Family Client.

    The LAN switch and the MSS Family Client have separate SNMP agents. In order to manage the combined functions of the LAN switch and the MSS Family Client, both agents must be configured.

Appendix B. 8371 LEC QoS Parameters

The following parameters are used to set the QoS:

all default values Use this option to set the QoS parameters to default values. In the following parameters the default values are also listed.

max burst size Sets the desired maximum burst size in frames for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the Maximum Burst Size traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired Maximim Burst Size traffic parameter for Data Direct VCCs. When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired Maximum Burst Size of that LEC is used for the Data Direct VCC. If both LECs request a reserved bandwidth connection, then the maximum of the desired Maximum Burst Sizes for the LE Clients is used for the Data Direct VCC. In any case (negotiation or not), the Maximim Burst Size is signaled only when SCR (sustained cell rate) is signaled. Although this parameter is expressed in units of cells, it is configured as an integer multiple of the Maximum Data Frame Size (specified in the LEC's C3 parameter) with a lower bound of one. Dependencies: This parameter is applicable only when traffic type is reserved bandwidth.

max reserved bandwidth The maximum reserved bandwidth acceptable for a Data Direct VCC. This parameter applies to both Data Direct VCC calls received by the LE Client and Data Direct VCC calls placed by the LE Client. For incoming calls, this parameter defines the maximum acceptable SCR for a Data Direct VCC. If SCR is not specified on the incoming call, then this parameter defines the maximum acceptable peak cell rate (PCR) for a Data Direct VCC with reserved bandwidth. Calls received with traffic parameters specifying higher rates will be released. If SCR is specified on the incoming call, the call will not be rejected due to the PCR of Maximum Burst Size. The constraint imposed by this parameter is not applicable to best_effort connections. For outgoing calls, this parameter sets an upper bound on the amount of reserved bandwidth that can be requested for a Data Direct VCC. Therefore, the traffic type and sustained cell rate parameters are dependent upon this parameter.

traffic type The desired traffic type for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the type of calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired type of traffic characteristics for Data Direct VCCs. When QoS parameters are negotiated, if either the source or target LEC desires a reserved bandwidth connection and both LECs support reserved bandwidth connections (that is, max reserved bandwidth > 0), then an attempt will be made to establish a reserved bandwidth Data Direct VCC between the two LECs. Otherwise, the Data Direct VCC will be a best-effort connection. Dependencies: max reserved bandwidth.

validate pcr of best effort vccs Whether or not to validate the peak cell rate of best-effort VCCs. When FALSE, best-effort VCCs will be accepted

without regard to the signaled forward PCR. When TRUE, best-effort VCC will be rejected if the signaled forward PCR exceeds the line rate of the LE Client ATM device. Calls will not be rejected due to the backward PCR. The signaled backward PCR will be honored if it does not exceed the line rate; otherwise, transmissons to the caller will be at line rate. **Note**: 1. Accepting best-effort VCCs with forward PCRs that exceed the line rate can result in poor performance due to excessive retransmissions; however, rejecting these VCCs can result in interoperability problems. 2. The yes setting is useful when callers will retry with a lower PCR following call rejection due to unavailable cell rate.

sustained cell rate The desired sustained cell rate for Data Direct VCCs. If QoS prameters are not negotiated, then this parameter specifies the SCR traffic parameter for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired SCR traffic parameter for Data Direct VCCs. When a reserved bandwidth VCC is negotiated and only one of the LE Client requests a reserved bandwidth connection, then the desired SCR of that LEC is used for the Data Direct VCC (subject to the upper bound imposed by the max reserved bandwidth parameter of the other LEC). If both LECs request a reserved bandwidth connection, then the maximum of the desired SCRs of the LE Clients is used for the Data Direct VCC (subject to the upper bound imposed by the max reserved bandwidth parameters of both LECs). In any case (negotiation or not), if the SCR that is to be signaled equals the PCR that is to be signaled, then the call is signaled with PCR only. Dependencies: max reserved bandwidth, traffic type and peak cell rate. This parameter is applicable only when the traffic type is reserved bandwidth.

qos class The desired QoS class for reserved bandwidth calls. If QoS parameters are not negotiated, then this parameter specifies the QoS class to be used for reserved bandwidth Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the QoS Class that is desired for Data Direct VCCs. Unspecified QoS Class is always used on best-effort calls. Specified QoS classes define objective values for ATM performance. Specified QoS Classes define objective values for ATM performance parameters such as cell loss ratio and cell transfer delay. The UNI Specification states that:

- **Specified QoS Class 1** shoud yield performance comparable to current digital private line performance.
- **Specified QoS Class 2** is intended for packet-switched video and audio in teleconferencing and multimedia applications.
- **Specified QoS Class 3** is intended for interoperation of connection-oriented protocols, such as frame relay
- **Specified QoS Class 4** is intended for interoperatoin on connectionless protocols, such as IP or SMDS

peak cell rate The desired peak cell rate for Data Direct VCCs. If QoS parameters are not negotiated, then this parameter specifies the PCR traffic parameters for Data Direct VCC calls placed by the LE Client. Otherwise, if QoS parameters are negotiated, this parameter specifies the desired PCR traffic parameter for Data Direct VCCs. The minimum

of the desired PCRs of the two LECs is used for negotiated best-effort VCCs. When a reserved bandwidth VCC is negotiated and only one of the LE Clients requests a reserved bandwidth connection, then the desired PCR of that LEC is used for the Data Direct VCC subject to the upper bound imposed by the line rate of the local ATM device. If both LECs request a reserved bandwidth connection, then the maximum of the desired PCRs of the LE Clients is used for the Data Direct VCC subject to the upper bound imposed on the line rate of the local ATM device.

negotiate qos Enable QoS parameter negotiation for Data Direct VCCs. This parameter should be enabled only when connecting to an IBM MSS. When this parameter is yes, the LE Client wil include an IBM Traffic Parameter TLV in the LE_JOIN_REQUEST and LE_ARP_RESPONSE frames sent to the LES. This TLV will include the values of max reserved bandwidth, traffic type, peak cell rate, sustained cell rate, max burst size and qos class. An IBM Traffic Parameter TLV may also be included in a LE_ARP_RESPONSE received by the LE Client, then the local configuration parameters must be used to set up the Data Direct VCC. If a TLV is included in a LE_ARP_RESPONSE, the LE Client must compare the contents of the TLV with the corresponiding local values to determine the "negotiated" or "best" set of parameters acceptable to both parties before signaling for the Data Direct VCC.

Appendix C. Special Notices

This publication is intended to help people involved in the design, construction, management and operation of ATM networks and specifically of those using the IBM MSS family of products. The information in this publication is not intended as the specification of any programming interfaces that are provided by the IBM MSS family. See the PUBLICATIONS section of the IBM Programming Announcement for the IBM MSS Server, IBM MSS Client, IBM MSS Domain Client, IBM 8270 Nways Token-Ring LAN Switch and IBM 8371 Multilayer Ethernet Switch in particular for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 USA.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

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

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating

environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX
AS/400
IBM
OS/2
PowerPC 603
P2P
S/390
System/390
APPN
BookManager
Nways
Power PC 603
PowerPC 603e
RS/6000
SP
VTAM

The following terms are trademarks of other companies:

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

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

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

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

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

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

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

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

Appendix D. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

D.1 International Technical Support Organization Publications

For information on ordering these ITSO publications see “How to Get ITSO Redbooks” on page 223.

- *MSS Release 2.1, Including the MSS Client and Domain Client*, SG24-5231
- *Understanding and Using MSS Release 1.1 and 2.0*, SG24-2115
- *Understanding and Using the IBM MSS Server*, SG24-4915

D.2 Redbooks on CD-ROMs

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

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

D.3 Other Publications

These publications are also relevant as further information sources:

- *Multi-Protocol Over ATM Version 1.0, AF-MPOA-0087, July 1997*

  The ATM Forum Worldwide Headquarters
  2570 West El Camino Real, Suite 304
  Mountain View, CA 94040-1313

- *Multiprotocol Switched Services (MSS) Server Introduction and Planning Guide*, GC30-3820
- *Multiprotocol Switched Services (MSS) Server Service and Maintenance Manual*, GY27-0354
- *Nways Multiprotocol Switched Services Server Interface Configuration and Software User’s Guide*, SC30-3818
- *Nways Multiprotocol Switched Services Configuring Protocols and Features Volume 1*, SC30-3819

- *Nways Multiprotocol Switched Services Configuring Protocols and Features Volume 2*, SC30-3994
- *8270 Switch Planning and Installation Guide*, GA27-4145
- *8272 LAN Switch Module Planning and Installation Guide*, GA27-4163
- *Nways Multiprotocol Switched Services Family Clients Interface Configuration and Software User's Guide*, SC30-3966
- *8371 Networking Multilayer Ethernet Switch Installation and Planning Guide*, GA27-4226
- *Event Logging System Messages Guide*, SC30-3682
- *8265 Nways ATM Switch User's Guide*, SA33-0456
- *Systems Network Architecture Advanced Peer-to-Peer Networking High Performance Routing Architecture Reference*, SV40-1018 (available only in softcopy on CD-ROM SK2T-6012)

How to Get ITSO Redbooks

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

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

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

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

- **E-mail Orders**

  Send orders via e-mail including information from the redbooks fax order form to:

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

- **Telephone Orders**

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

- **Fax Orders**

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

This information was current at the time of publication, but is continually subject to change. The latest information for customer may be found at http://www.redbooks.ibm.com/ and for IBM employees at http://w3.itso.ibm.com/.

IBM Intranet for Employees

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

IBM Redbook Fax Order Form

Please send me the following:

Title	Order Number	Quantity

First name	Last name	
Company		
Address		
City	Postal code	Country
Telephone number	Telefax number	VAT number

☐ Invoice to customer number _____

☐ Credit card number _____

Credit card expiration date	Card issued to	Signature

We accept American Express, Diners, Eurocard, Master Card, and Visa. Payment by credit card not available in all countries. Signature mandatory for credit card payment.

List of Abbreviations

APA	all points addressable
APPN	advanced peer-to-peer networking
ARP	address resolution protocol
ASRT	adaptive source-route transparent
ATM	asynchronous transfer mode
BBCM	bridging broadcast manager
BCM	broadcast manager
BUS	broadcast and unknown server
CIP	Classical IP
CPSW	control point switch
DLSw	data link switching
DMA	direct memory access
DPF	dynamic protocol filter
DLL	data link layer
ELAN	emulated LAN
ELS	event logging system
EPROM	erasable programmable read-only memory
ESI	end system identifier
FTP	file transfer program
HPR	high performance routing
HTML	Hypertext Markup Language
IBM	International Business Machines Corporation
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IPX	internetwork packet exchange
ITSO	International Technical Support Organization
LAN	local area network
LANE	LAN Emulation
LEC	LAN Emulation Client
LECS	LAN Emulation Configuration Server
LES	LAN Emulation Server
LIS	logical IP subnet
LSI	LAN Emulation Shortcut Interface
MAC	media access control
MPC	MPOA Client
MPOA	multiprotocol over ATM
MPS	MPOA Server
MSS	Multiprotocol Switched Services
MTU	maximum transmission unit
NBMA	non-broadcast multi-access
NHC	NHRP client
NHRP	Next Hop Resolution Protocol
NHS	NHRP Server
OSPF	open shortest path first
PNNI	private node-to-node interface
PTF	program temporary fix
PVLAN	protocol VLAN
QoS	Quality of Service
RFC	request for comments
RSVP	Resource Reservation Protocol
SDU	service data unit
SRAM	static random-access memory
SNA	systems network architecture
SRB	source-route bridge
SVC	switched virtual circuit
TCP/IP	transmission control protocol/internet protocol
TFTP	trivial FTP
TG	transmission group
TLV	type/length/value
TOS	type of service
VCC	virtual circuit connection
VLAN	virtual LAN
VNI	virtual network interface
WAN	wide area network

Index

Numerics

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

V

W

ITSO Redbook Evaluation

Layer 3 Switching Using MSS
and MSS Release 2.2 Enhancements
SG24-5311-00

Your feedback is very important to help us maintain the quality of ITSO redbooks. **Please complete this questionnaire and return it using one of the following methods:**

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

Which of the following best describes you?
_ **Customer** _ **Business Partner** _ **Solution Developer** _ **IBM employee**
_ **None of the above**

Please rate your overall satisfaction with this book using the scale:
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction _____

Please answer the following questions:

Was this redbook published in time for your needs? Yes___ No___

If no, please explain:

_____

_____

_____

_____

What other redbooks would you like to see published?

_____

_____

_____

Comments/Suggestions: **(THANK YOU FOR YOUR FEEDBACK!)**

_____

_____

_____

_____

_____

SG24-5311-00

Printed in the U.S.A.

Layer 3 Switching Using MSS and MSS Release 2.2 Enhancements

SG24-5311-00