Novell NetWare® 6

www.novell.com

SERVER COMMUNICATIONS
ADMINISTRATION GUIDE

Novell®

Legal Notices

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

Server Communications Administration Guide
October 2001
103-000145-001

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

Internetwork Packet Exchange and IPX are trademarks of Novell, Inc.

NE2000 is a trademark of Novell, Inc.

NetWare is a registered trademark of Novell, Inc., in the United States and other countries.

NetWare Core Protocol and NCP are trademarks of Novell, Inc.

NetWare/IP is a trademark of Novell, Inc.

NetWare Loadable Module and NLM are trademarks of Novell, Inc.

NetWare Link Service Protocol and NLSP are trademarks of Novell, Inc.

Novell is a registered trademark of Novell, Inc., in the United States and other countries.

Novell Directory Services and NDS are registered trademarks of Novell, Inc., in the United States and other countries.

Packet Burst is a trademark of Novell, Inc.

Third-Party Trademarks

All third-party trademarks are the property of their respective owners.

Contents

Server Operating System Communications

NetWare® has traditionally used IPX™ and its protocols for network communication. NetWare 4 supported IP networks through NetWare/IP™. The release of NetWare 6 allows a choice of running networks with just IPX, with both IP and IPX, or with pure IP.

The Internet Protocol comprises a set of publicly available protocols that provides the means by which computers communicate on the Internet.

IPX comprises a set of protocols that facilitate communication between computers on NetWare networks.

In order to make IP run on NetWare, the public protocols of IP had to be incorporated into, and replace, the proprietary protocols in NetWare. Since NDS® is the heart of NetWare, it was used to bring all the Internet Protocols together in NetWare 6. This makes it possible to configure and maintain the protocols using NetWare Administrator.

Compatibility Mode (CM) (page 31) maintains backward compatibility with IPX NetWare systems. You can install a server or client using one of three methods: IP (with compatibility mode), IPX, or both IP and IPX. CM provides translation between IP and IPX by recognizing IPX packets and then determining how to forward them. A Migration Agent (MA) (page 32) on the server uses CM to bridge IP and IPX networks while maintaining protocol purity on each of the respective networks.

No other networking software vendor has combined Internet protocols with a proven networking platform to provide a pure IP networking solution. Novell® has accomplished this because of the flexibility and scalability inherent in NDS. By extending the NDS schema and adding objects, Novell has built a network operating system out of publicly available Intenet Protocols and made it work with IPX for backward compatibility. The Server

Operating System Communications documentation outlines and contrasts the protocols used in Novell's IP and IPX implementations of NetWare 6.

Documentation Conventions

In this documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

Also, a trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

1 Understanding Network Communications

Network communications involve many complex operations, but these operations can be grouped into four major categories:

- Identifying Devices (page 12)

  For computers to communicate on networks, each must have an address. Just as postal services are unable to deliver a package without an address, computers are unable to communicate without an address. Since computers use numbers for addresses, but humans have an easier time distinguishing names, computers use protocols to match the number address to a name.

- Finding Services (page 13)

  Once a computer has an address and/or name, it can start communicating with other computers. Its first communication is to let other computers know what services it has to offer. Then it must find out what services are being offered by other computers on the network. This is accomplished by using one of several service advertising and location protocols.

- Moving Packets (page 13)

  Having discovered other computers' addresses and the services they offer, a computer can start moving packets between itself and other hosts. To communicate efficiently, though, computers must know the fastest way to move data from point A to point B. Computers determine the best route from computer to computer with routing protocols.

- Coordination (page 14)

  Finally, network communication depends upon maintaining data integrity. NetWare® servers must coordinate time and NDS® replicas to ensure data integrity on the network. Time servers coordinate their time

with other servers and relay the correct network time to NetWare clients. NDS replication is similar to time synchronization in that servers must keep and share accurate information to maintain fault tolerance and distributed access to the database.

Identifying Devices

Devices on networks must be uniquely identified so that other devices can find and use their services. Since IPX Addressing (page 16) was designed to be simple and require little maintenance, it doesn't rely on protocols to enhance its functionality.

IP Addressing (page 14) and IP Subnetting (page 15) are both more complex, however, and require maintenance type protocols to make administration manageable. ARP (page 16) and DHCP (page 17) are two commonly used IP addressing protocols.

Two kinds of addresses identify hosts on the network: hardware or media access control (MAC) addresses, and software addresses. IPX™ uses the MAC address of the Ethernet or token ring network board to identify the host on the network. Since the MAC and node addresses are the same, there is no further translation required to identify the host. IP addresses are not the same as the MAC address of the network board, so IP addresses must be translated into MAC addresses. Address Resolution Protocol (ARP) translates IP addresses to MAC addresses on IP networks.

Dynamic Host Control Protocol (DHCP) is an Internet protocol that provides dynamic distribution of IP addresses to workstations. DHCP helps network administrators with the task of assigning IP addresses to workstations and lessens the problems associated with a shortage of IP addresses. There is no equivalent to DHCP in IPX networks because of the abundance of IPX addresses and their ability to use the MAC address as the software address.

Finding Services

Once a computer is uniquely identified on the network, it can let other computers know what services it offers, or it can request services from another computer. There are three service protocols that maintain lists of computers and the services they offer:

Protocol Name	Protocol Type
Service Location Protocol (SLP)	IP
Service Advertising Protocol (SAP)	IPX
Domain Name Service (DNS)	IP

SLP (page 17), an Internet protocol, and SAP (page 24), an IPX protocol, are both used to locate and advertise network services.

DNS (page 24) is an Internet standard service that provides IP address-to-host name resolution. Its primary purpose is to match the name of a computer, such as host1.novell.com, with its IP address. DNS can also map certain Internet server services, such as E-mail and Web, to specific hosts.

Host files can also be used on private networks to accomplish IP address-to-host name resolution.

Moving Packets

Computers use and provide services by exchanging packets. Packet exchange can be accomplished only if the computers know how to move information amongst themselves. Computers learn the path, or route, to other computers by using routing protocols such as the following:

- OSPF (page 25)
- NLSP (page 25)
- RIP, RIP II (page 25)

There are two kinds of routing protocols, distinguished by their mode of best route discovery:

- Distance Vector
- Link State

Distance vector routing protocols determine the best route from one computer to another based on the distance, or number of hops, and the time, or ticks, that separate hosts. Link state routing protocols use a cost metric to determine the best path between hosts.

Link state routing protocols are generally more accurate and efficient than distance vector routing protocols and are better suited for traversing WAN links. The table below shows the protocol and routing types associated with the routing protocols:

Routing Protocol	Protocol Type	Routing Type
OSPF	IP	Link State
NLSP	IPX	Link State
RIP	IP and IPX	Distance Vector
RIP II	IP	Distance Vector

Open Shortest Path First (OSPF) is a link state IP routing protocol. Its IPX equivalent is NetWare Link Service Protocol™ (NLSP™). Routing Information Protocol (RIP) is a distance vector routing protocol used for both IP and IPX routing, but with some variation between protocols. RIP II is a newer IP routing protocol based on RIP that adds support for a subnet mask.

Coordination

See “Time Synchronization” on page 26 and “NDS Replication” on page 27.

IP Addressing

The IP address for a node is a logical address, independent of any particular hardware, network topology, or media type. The IP address is a 4-byte (32-bit) numeric value that identifies both a network and a local host or node (computer or other device) on that network. The 4-byte IP address is usually represented in dotted decimal notation. Each byte is represented by a decimal number, and periods separate the bytes, for example, 129.47.6.17.

A conflict arises with Ethernet networks, because IP uses a 32-bit address and Ethernet uses a 48-bit Ethernet address. To associate the IP address to a physical address on an Ethernet network, a mapping must occur between the

two types. The Address Resolution Protocol (ARP) maps the IP address to the physical address. ARP mapping is limited to networks that support hardware broadcast.

IP Subnetting

One IP network can be divided into smaller networks, called subnets. The following are reasons to divide your network:

- Use multiple media—It can be impossible, inconvenient, or too expensive to connect all nodes to a single network medium when these nodes are too far apart or already connected to different media.
- Reduce congestion—Traffic between nodes on a single network uses network bandwidth. As a result, more bandwidth is required when you have more nodes. Splitting a network reduces the number of nodes on a data-link network. Fewer nodes generate less traffic and, as a consequence, less congestion.
- Reduce processor use—Because each node on a network must react to every broadcast, reducing the number of nodes reduces processor use and congestion.
- Isolate a network—By splitting a large network into small networks, you limit the impact of one network's problems on another. Such problems can include network hardware failures, such as an open Ethernet tap, or software failures, such as a broadcast storm.
- Improve security—On a broadcast network medium such as Ethernet, each node on a network has access to all packets sent on that network. By enabling sensitive network traffic on only one network, other network monitors can be prevented from accessing this sensitive traffic.
- Make efficient use of IP address space—If you are using a Class A or B network number and have multiple small physical networks, you can divide the IP address space into multiple IP subnets and assign them to individual physical networks. Another option is to obtain several Class C network numbers, although this is less desirable.

IPX Addressing

IPX defines its own internetwork and intranode (or intranetwork) addressing. For intranode addressing, IPX uses the physical address assigned to the network board. The IPX network address uniquely identifies an IPX server on an IPX network and individual processes within the server. A complete IPX network address is a 12-byte hexadecimal number comprising the following components:

- A 4-byte network number (server)
- A 6-byte node number (server)
- A 2-byte socket number (server process)

The following is an example of a complete IPX network address:

```
FEDCBA98 1A2B3C5D7E9F 0453
```

Each number in an IPX address is contained in a field in the IPX header and represents a source or destination network, node, or socket. The network number is used only for network-layer operations, namely routing. The node number is used for local, or same-segment, packet transmission. The socket number directs a packet to a process operating within a node.

ARP

Unlike IPX, IP addresses are not the same as the hardware address of the network board, so there must be a way to discover the physical, or media access control (MAC) address. The Address Resolution Protocol (ARP) performs this task. When an IP address is mapped to a MAC address, ARP is used on broadcast networks such as Ethernet, token ring, and ARCnet. When a node uses IP to send a packet, it must determine which physical address on the network corresponds to the destination IP address. To find the physical address, the node broadcasts an ARP packet containing the destination IP address. The node with the specified destination IP address sends its physical address back to the requesting node.To speed packet transmissions and reduce the number of broadcast requests that must be examined by every node on the network, each node keeps an address resolution cache, or ARP table. Each time the node broadcasts an ARP request and receives a response, it creates an entry in its address resolution cache. The entry maps the IP address to the physical address. When the node sends an IP packet, it looks up the IP address in its cache and uses the physical address, if found. The node broadcasts an ARP request only if the IP address is not in its cache.

DHCP

The Dynamic Host Configuration Protocol (DHCP) uses a client-server structure to provide configuration parameters to hosts. DHCP consists of a protocol for providing host-specific configuration parameters from a DHCP server (or collection of DHCP servers) to a host and a mechanism to allocate network addresses to a host.

When the DHCP server is loaded, it reads its configuration information from NDS and stores the information in its cache. As the DHCP server assigns addresses to clients, it updates NDS, adding IP address objects or modifying their NDS status information. The DHCP server can be configured to maintain an audit log of this activity.

The administrator can use the DNS/DHCP Administration utility to view objects to see how addresses have been assigned.

SLP

The Service Location Protocol provides the same function in IP networks as SAP provides in IPX networks. It registers information in a database and allows clients to query the database to find services. There are, however, two principal differences between SAP and SLP:

- SLP does not maintain a global database of services. It registers services only in the local area. It discovers services in the local area via multicast requests, which are forwarded using NDS replication from network to network within a site.
- SLP assumes that the client is able to locate either services themselves, or a database server representing those services, using these pan-network multicasts.

Through Novell's integration of SLP with NDS, local SLP information is compiled to provide a global representation of all available services on the network. This provides dynamic discovery of services locally and scalability in large networks.

The topics listed below explain the components of SLP:

- "How SLP Works" on page 20
- "SLP Agents" on page 18
- "SLP NDS Objects" on page 22

SLP Agents

The three types of agents that NetWare 5 SLP uses are

- User agents, which acquire service handles for user applications
- Service agents, which advertise service handles
- Directory agents, which collect service handles in internetworked enterprises

Applications running on a computer are represented by a User agent that understands the service and resource needs of the application. Each network service is represented by a Service agent, which makes it available to user agents. SLP dynamically maintains service attributes, so that a User agent can obtain current information.

Of the agents, the Directory Agents (page 18) have the largest role in NDS SLP.

Directory Agents

The point of interface between SLP and NDS is the SLP Directory agent. The Directory agent is a common data storage of network service information collected through SLP. The Directory agent uses NDS as its database for network service information that is distributed globally. NDS adds significant value to SLP by leveraging existing NDS standards for configuring NDS tree structures, for a central point of administration, and for the ability of NDS to replicate service information.NDS replication services allow Directory agent-to-Directory agent communication. This is unique in SLP implementations and it facilitates global distribution of SLP database information. NDS replica services give the Directory agent the ability to access global services from a local replica. The Directory agent is responsible for processing the following SLP protocol messages:

- Service Registration
- Service Deregistration
- Service Request
- Service Type Request
- Attribute Request

These SLP protocol messages either enter, delete, or query information in the Directory agent's service database.

Service Registration

A Service agent forwards all known services to the Directory agent using a service register request. The register contains the URL, attributes, language indicator, and a time to live (lifetime). The service registration occurs when attributes are being updated or modified and once every lifetime period.

Service Deregistration

A Service agent sends a service deregister request to the Directory agent when the service is no longer available.

Service Request

A User agent sends a service request to the Directory agent when it is looking for services. The Directory agent returns only those services with a valid lifetime. The User agent might filter services by providing a predicate list. The Directory agent must filter services when the predicate list is supplied.

Service Type Request

A user agent sends a service type request to the Directory agent when it is looking for all service types or all service types within a specific name authority.

Attribute Request

A User agent sends an attribute request to the Directory agent either for a specific URL or for a group of URLs specified by the service type.

Novell's Enhancements to SLP

As mentioned previously, once the SLP service information has been stored in the NDS tree, the normal replication and distribution processing of NDS will guarantee its global accessibility. Only those Directory agents granted access rights to the Scope container object will have access to the SLP service information in that scope.To reduce bandwidth requirements on large networks, the NetWare SLP Directory agent doesn't use IP multicast. In a small network, IP multicast is a viable technology that can be coupled with the SLP user and Service agents to provide acceptable discovery service. As the network expands, the IP multicast can cause some bandwidth reduction, as

routers must forward the multicast packets to all registered nodes. To solve this problem, NetWare SLP Directory agents collect the information from local segments and then establish IP unicast relationships. Although the SLP RFC defines the Directory agent and its relationship to the user and Service agents, the specification doesn't address the relationship among multiple Directory agents. A Directory Agent-to-Directory Agent protocol is mentioned in the specification, but the work has been left to a future version of the RFC. NDS, however, provides a solution. The NDS replicated database can provide authenticated and synchronized information across networks while preserving network bandwidth.

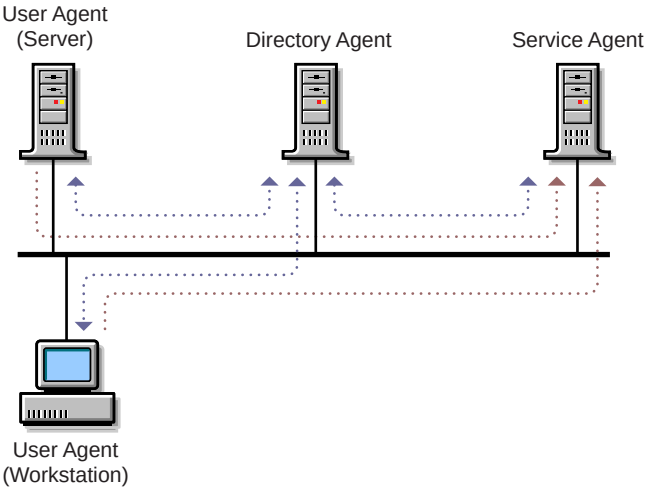
Scope Container Object

SLP employs the Scope container object which defines a logical grouping of services. The Scope object allows network administrators to logically group services according to geographical, geo-political, service type, or any other administrative criteria in order to control distribution or visibility on the network. The primary goal of the SLP Scope is to enhance the scalability of gathering and distributing network service information.

How SLP Works

The following figure illustrates how SLP registers a service provider on a local segment. Each agent must register its own services. Whether the User agent is on the server or on a workstation, it can register as a client after it communicates with the Directory agent to see what services are available. Once the service is registered with the Directory agent or Service agent, you can register or deregister the service.

Figure 1 Service Location Protocol

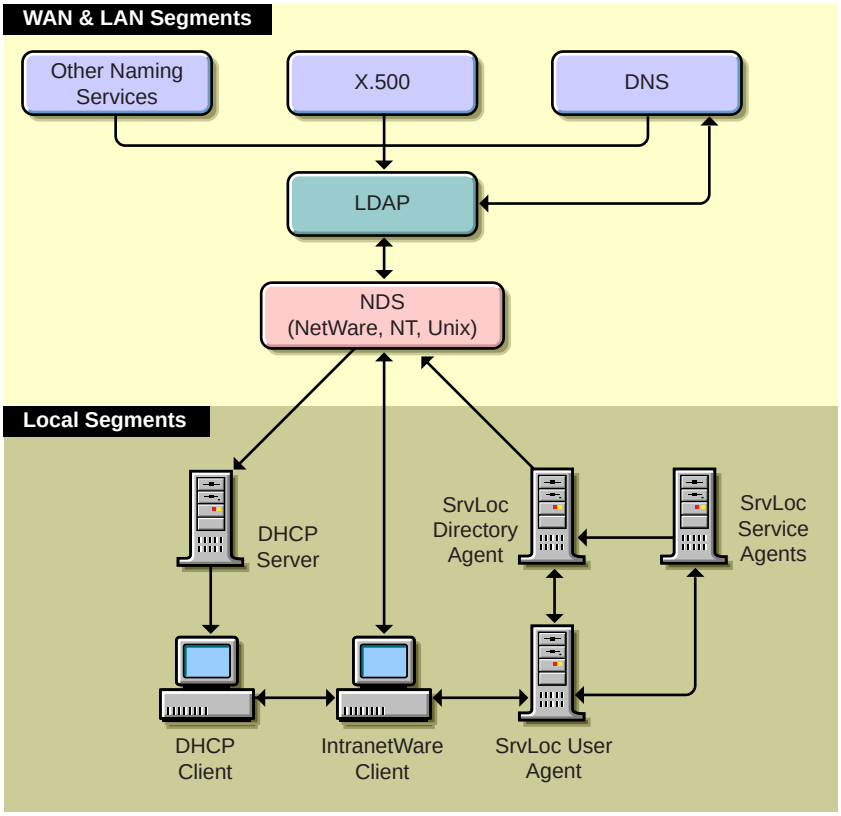


Once the application has registered with the SLP User agent, it can look up a service or get a list of services and read the attributes of a service, using either blocking calls or synchronous calls. In the IP environment, this information is pulled out of the Directory agent and put into NDS so that users and administrators can know what services are available in a local area, provided the proper security rights are granted.

A Novell client can use the User agent to go into an SLP Directory agent or Service agent, or into NDS to reach out to other LAN or WAN segments, as shown in Figure 2 on page 22.

This method does not rely on service information obtained from routers. Instead, NDS is used for global communication of information. Through this method, service updates on local segments are just as reliable and dynamic as on IPX SAP-based networks.

Figure 2 Integrated Network Services Discovery



SLP NDS Objects

Following are the NDS objects represented by SLP:

- Scope container object
- SLP Service object
- SLP Directory Agent object

The SLP Scope container object represents an SLP scope and is the container in which SLP Service objects are stored.

SLP Service objects represent a network service discovered through the Service Location Protocol. They contain all of the SLP information about the network service, including its network address and attributes.

The SLP Directory Agent object represents an SLP Directory agent.

SLP Scope Container Object

The SLP Scope container object is the storage container for SLP service information. Each object contains all the SLP Service objects for the specific scope. The NDS administrator can replicate the container into other partitions within the tree or within federated trees. The object is a stand-alone entity within the NDS tree and there is no relationship between its distinguished name, the tree name, and the scope name. When a Service agent forwards a service record to a Directory agent within a specific scope, the scope name is mapped to the Scope object by using the name attribute within the container object. The SLP Scope object must contain rights to read, write, and browse the container because the access rights of the Directory Agent object access are equivalent to the access rights of the Scope object.Because the Scope object uses distinguished name syntax, the Scope object can be moved to a different location in the tree and NDS will automatically change all values to reflect the new location.

SLP Service Object

The SLP Service object is a leaf object that represents a service registration. SLP Service objects are subordinate to the SLP Scope object and contain all information supplied by a service registration. SLP Service objects are stored in the appropriate SLP Scope object according to their scope.

Directory Agent Object

The SLP Directory Agent object is a leaf object that represents a single instance of a Directory agent. Multiple Directory agents cannot share a single object. This object defines the Directory agent's configuration, scope, and security. The Directory agent uses this object to log in to the server and operate under the access control requirements assigned to the NCP Server object.

NCP Server Object

The NetWare installation program creates an NCP_SERVER object for every server within the tree. The Directory agent adds an attribute to the NCP_SERVER class definition called SLP Directory Agent DN. The SLP Directory Agent DN contains the distinguished name of the Directory Agent object. It is used as a pointer from the NCP Server object to the Directory Agent object.

SAP

The Service Advertising Protocol provides the same function in IPX networks as SLP in IP networks. It registers information in a database and allows clients to query the database to find services.NetWare servers using IPX use SAP to advertise their services and network addresses. Routers gather this information and share it with other routers. Workstations on the network access the information provided by routers to determine which services are available on the network and to obtain the IPX address of the services. Workstations use this information to initiate a session with a service.SAP makes the process of adding and removing services on an internetwork dynamic. As servers start up, they use SAP to advertise their services; as they are brought down, they use SAP to indicate that their services are no longer available.As a router becomes aware of any change in the internetwork server layout, this information is broadcast immediately to all neighboring routers. SAP broadcast packets containing all server information known to the router are sent periodically—the default is every 60 seconds. These broadcasts keep all routers on the internetwork synchronized and provide a means of updating routing information when a router or server has become inaccessible since the last broadcast. A server might be inaccessible because a router went down, or because a router dropped a packet containing a notification that the route to the server is unreachable. Servers that are inaccessible do not appear in the SAP broadcast.

DNS

The Domain Name System (DNS) is a distributed database system that provides hostname-to-IP resource mapping (usually the IP address) and other information for computers on an internetwork. Any computer on the Internet can use a DNS server to locate any other computer on the Internet.

DNS is made up of two distinct components:

- The DNS hierarchy specifies the structure, naming conventions, and delegation of authority in the DNS service.
- The DNS name service provides the actual name-to-address mapping mechanism.

OSPF

Open Shortest Path First (OSPF) is an IP link state routing protocol. Link state routers exchange information about the state of their network connections or links. Using this information, each router can construct the topology of the internetwork, and from that derive a routing table consisting of the most efficient paths between devices.

OSPF offers the following advantages over IP RIP:

- Faster convergence of router information tables
- First hand routing information
- Generates less traffic
- No count-to-infinity problem

NLSP

NLSP is an IPX link state routing protocol that was developed to respond to limitations that arise when implementing IPX RIP and SAP in larger internetworks, particularly over WAN links. Link state routers exchange information about the state of their network connections or links. Using this information, each router can construct the topology of the internetwork and derive routing information.

NLSP offers the following advantages over IPX RIP

- Faster convergence of router information tables
- First hand routing information
- Generates less traffic

RIP, RIP II

RIP is a distance vector routing protocol used for both IP and IPX routing, but with slightly different implementations. IP RIP and IPX RIP use similar processes for discovering, maintaining, and prioritizing routes. They both send route requests for obtaining routing information and send periodic route updates to make sure the routing information tables are synchronized. The major differences between IP RIP and IPX RIP are the protocols they are associated with, the way they prioritize routes, and the routing table update interval.

RIP II is an IP routing protocol that includes the following enhancements over RIP:

- Provides a password for authentication
- Allows specification of a subnet mask
- Allows multicast addressing

Time Synchronization

Synchronizing time across the network provides a service that maintains consistent time stamps for enterprise environments with several servers in different time zones.

Time synchronization provides network time for the following:

- File systems
- Messaging services
- Network applications

NDS Time Servers use TIMESYNC to provide synchronized time for network services.

If there are fewer than 30 servers on the network, use the default settings of a single reference time server and a secondary time server.

For more than 30 servers, plan a custom environment using a reference time server, primary time server, and secondary time servers. You can also specify which communication method the time servers will use: SAP or a configured list.

TIMESYNC allows NetWare servers to synchronize their time with an authoritative external time source, such as an atomic clock, through an asynchronous connection such as a modem. NTP (Network Time Protocol) is an Internet Protocol that can be used with TIMESYNC to query authoritative time servers over the Internet rather than with a dialup connection.

NDS Replication

NDS replica synchronization ensures that changes to NDS objects are synchronized among all replicas of the partition. This means that any server that holds a replica of a partition must communicate with the other servers to synchronize a change.

In NetWare 6, NDS replica synchronization is more efficient and produces less network traffic than previous versions, because instead of automatically synchronizing replicas, servers are queried to find out if they are synchronized or not. If a server is out of synchronization, the update is sent. But if the server is synchronized, there is no need to send the update.

2 Planning

As an open standard, pure IP offers flexibility and interoperability, now available to users of NetWare®. Although many customers may use both IP and IPX™ on their networks, a pure IP network is easier to administer and more easily integrated with other systems, such as UNIX* and Windows NT*. New IP migration tools and services in NetWare 6™ make migrating to a pure IP network managable, even for the largest networks. Whether you choose to migrate from IPX to IP will largely depend on the goals of your organization.One of the biggest advantages of migrating a network from IPX to IP is reduced administrative costs. Migrating from IPX to IP will be most beneficial for customers already supporting both protocols, and for those expending a significant portion of their Information Services (IS) budget managing IPX.

Migrating the network from IPX to IP is not necessary to take advantage of the increased connectivity of NetWare 6. If you are satisfied with your existing network infrastructure, but would like to make NetWare services available to IP clients, you can upgrade servers and clients to NetWare 6 and load both protocol stacks.

- If you are installing or upgrading a server, see NetWare 6 Installation Guide.
- To decide what protocol to use, see “Protocol Selection” on page 30.

Protocol Selection

Previous versions of NetWare used Internetwork Packet Exchange (IPX), a proprietary protocol developed by Novell®, for network communications. NetWare 6 uses TCP/IP (Transmission Control Protocol/Internet Protocol), IPX, or a combination of both IP and IPX.

With increasing access to worldwide data exchange through the Internet, IP has become so popular that many companies'networks now require it. But IPX and IP are two separate protocols. If you run both, you must maintain both. Administering routers, bridges, switches and other hardware components required for multiprotocol network communications can prove prohibitive.

Potentially, migrating from IPX to IP could prove costly, and there are some important considerations. Attempting to move a large number of servers and clients to NetWare 6 simultaneously won't generally be practical. It may be neccessary to introduce IP components over time, depending on the size and complexity of your network. You might choose to upgrade only servers as a preliminary phase, and later upgrade clients. To understand your migration options, see “Planning Migration” on page 30.

Although Novell now recommends the use of pure IP on your network, small, private networks might be more efficient using an IPX-only implementation. As you'll see in the discussion regarding addressing, IPX requires no special address resolution protocols—it can assign addresses dynamically, and addresses are abundant. IP is better suited for large IP-based networks attached to the Internet, to WAN links, or where IP is the exclusively required protocol. If you don't require IP for any of these reasons, and you can use a pure IPX network, you might find IPX implementation easier to administer.

Planning Migration

However you choose to migrate from IPX to IP, the cost and difficulty usually associated with a major change such as this is offset by NetWare 6 migration tools designed to facilitate migration without loss of connectivity or IPX application support. The migration tools include the following:

- Compatibility Mode (CM) (page 31)
- Migration Agent (MA) (page 32)

These components can be loaded on the same server. The CMD runs on all NetWare 6 servers by default. Only one MA is required for each IPX network connected to the IP backbone.

Because most existing NetWare servers have some dependency on IPX applications and services, NetWare 6 installs with IP running Compatibility Mode as the default. When you install NetWare 6, you can choose to load with IPX only, or IP with Compatibility Mode. Although the Compatibility Mode driver is loaded on the server by default, it remains dormant until the server receives a request for IPX services. This allows backward compatibility with IPX while using minimal system resources.

Use of the other components of Compatibility Mode will be determined by your networking requirenment and existing infrastructure. Each component's role in a multiprotocol network is described to help you determine if you need to use it.

Compatibility Mode (CM)

The Compatibility Mode driver (CMD) has two parts, one for the server and one for the client. At the server, the CMD is viewed as a network adapter. You can bind both protocols to the CMD and it acts like a router when IPX packets need to be sent within the server. Otherwise, the CMD patiently waits in the background, doing nothing and using no resources.

At the workstation, the CMD is invisible because it is an integral part of the new client. It provides the IP communications link required by an IP client. Because NetWare 6 is pure IP, there is no need for IPX at the client.

The IPX Compatibility driver's job is to provide IPX connectivity over the IP network, allowing applications using the IPX stack for communications to function in an IP network. The IPX Compatibility driver also allows IP systems to communicate with IPX systems by using the services of Migration Agents.The IPX Compatibility driver treats the IP network as a virtual IPX network segment (CMD network segment), by encapsulating IPX datagrams inside UDP datagrams, and by resolving RIP and SAP requests through the use of the Service Location Protocol (SLP).

IPX Compatibility Feature Dependencies

If you want to run IPX applications in your IP network, or you need to connect IP systems with IPX systems, you must ensure that the Service Location Protocol is enabled across the networks, because the IPX Compatibility drivers are dependent on the capabilities of SLP. Customers who want to interconnect IPX and IP systems must introduce at least one Migration Agent on the network.

The Virtual IPX Network Created for the IPX Compatibility Feature

The default IPX network number (CMD network number) assigned to the virtual IPX network created by the IPX Compatibility drivers is 0xFFFFFFFD. If you want to set up Migration Agents to interconnect IP systems with IPX systems, you must ensure that the CMD network number does not conflict with the internal IPX network number of a server or with the IPX network number of a network segment. You must also ensure that IPX routers are not filtering this address.If you find a system or a segment that conflicts with the CMD network number, you have the option of overriding the default CMD network number by modifying the configuration of IP-only clients and servers. You might find it easier to change the network number of the conflicting system or segment, rather than trying to override the default CMD network number.

Migration Agent (MA)

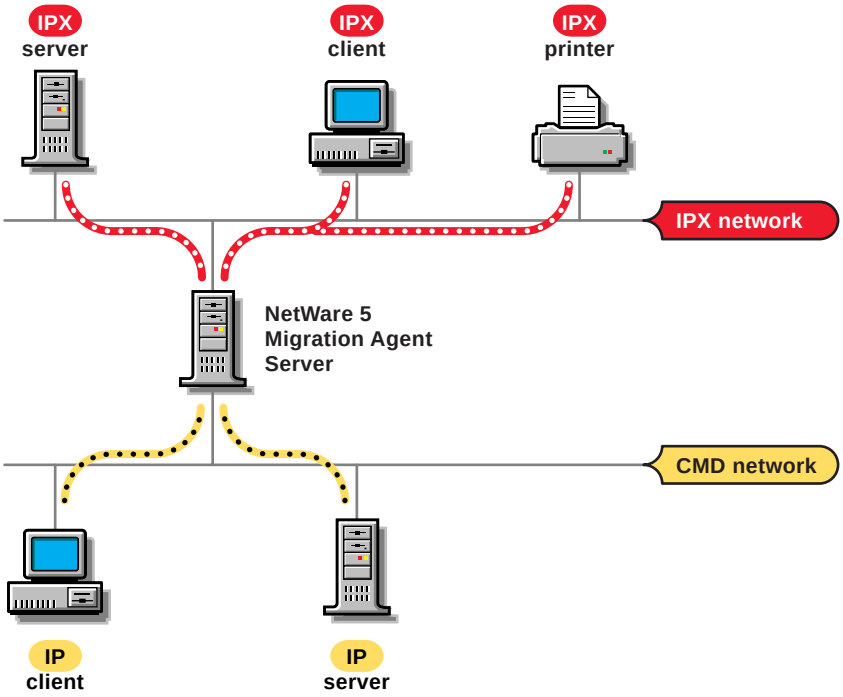
The Migration Agent is a tool that enables communication between IPX and IP systems. It also enables you to create an IP backbone that interconnects IPX segments. Use this tool when you want to migrate systems from IPX to IP in a phased manner without losing connectivity.

The MA takes the IPX requests, which are in an IP packet, and tunnels the IP packet in an IPX wrapper to be sent out on the IPX wire. The opposite occurs when an IPX packet is sent across the IP backbone.

In previous versions of NetWare, IP access was provided on NetWare networks with NetWare/IP™. NetWare/IP took every packet (all were IPX packets) and tunneled each in an IP wrapper. This kind of IP tunneling is no longer needed, because tunneling has been reversed in NetWare 6—instead of tunneling IP packets in IPX with NetWare/IP, IPX packets are now tunneled in IP packets with the Migration Agent. Now, only the few IPX requests require tunneling, providing better throughput and efficiency.

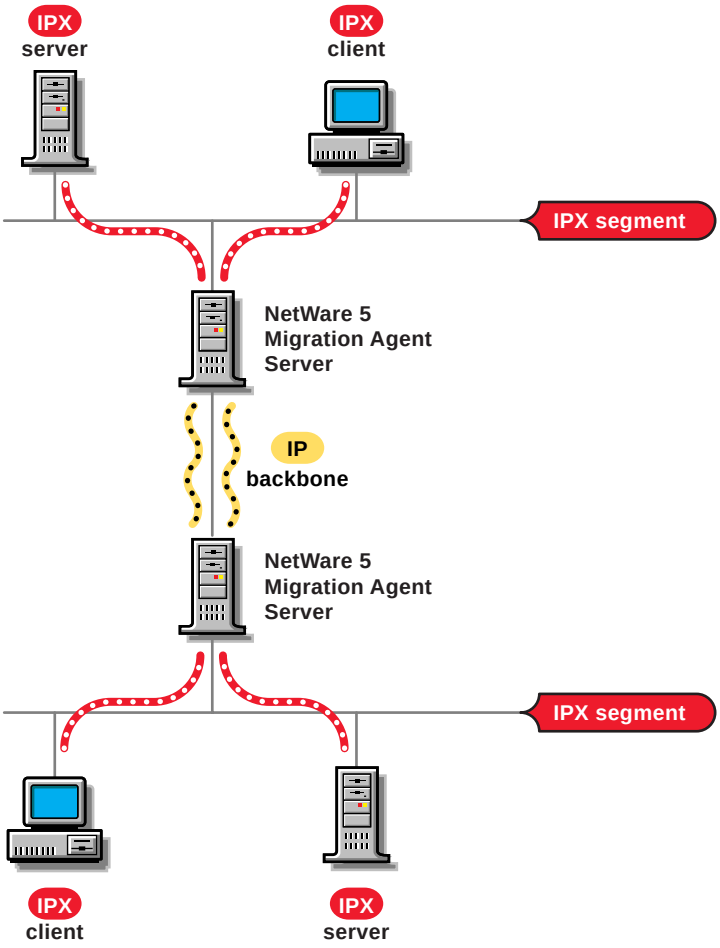
The MA serves as a router between the IPX network and the virtual IPX network segment created by the IPX Compatibility drivers as illustrated in Figure 3 on page 33.

Figure 3 Migration Agent Interconnecting IP and IPX Nodes



More than one MA is needed to enable resiliency and load-sharing, or when you want to interconnect IPX segments with an IP backbone. Figure 4 on page 34 shows two Migration Agents interconnecting two IPX segments.

Figure 4 Migration Agents Interconnecting IPX Segments



The MA is supported only at the NetWare server. The MA is enabled by loading the IPX Compatibility driver (scmd.nlm) with the Migration Agent option. The Migration Agents are then used by the IP systems on the network.If more than one MA is needed, all Migration Agents must be able to access the same IPX networks or be able to exchange IPX network information. Migration Agents exchange IPX network information by invoking the IP Backbone Support feature, which is accomplished by loading the scmd.nlm with the backbone support options.

Migration Agent Dependencies

To set up an MA, the Service Location Protocol must be enabled across the networks, because Migration Agents are dependent on the capabilities of SLP.

Dynamic Discovery of Migration Agents by IP Systems

The IPX Compatibility drivers are capable of dynamically discovering Migration Agents, but you can also choose to statically configure the address of the MA if more control is desired.The IPX Compatibility driver will discover an MA if it is in the same IP network, and will give preference to an MA within the local IP subnet. The address of the MA must be specified in IP systems that reside in different IP networks.The address of the MAcan be configured either by manipulating the local configuration files or by disseminating the information through DHCP.

Protocol Stack Options

The server and client connectivity capabilities are limited by the options selected when systems are installed. Systems can be installed with the following protocol options:

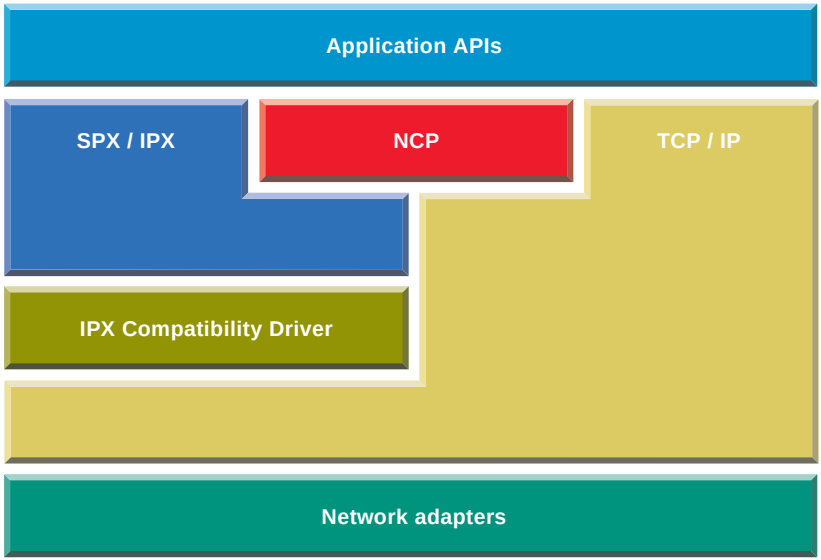
- IP Only
- IPX Only
- IP and IPX

The protocol install option determines the binding between protocol stacks and network adapters. It does not determine which protocol stacks are loaded in the system. For example, if the IP option is selected, only the TCP/IP stack is attached to the network adapter.

IP Install Option

Servers installed with IP alone have both the TCP/IP and the IPX stacks loaded, but only the TCP/IP stack is bound to the network adapter. (Systems installed with IP and IPX are configured to establish NCP connections over either the TCP/IP stack or over the IPX stack.) Figure 5 on page 36 shows the relationship between the protocol stacks when installing a system with IP only.

Figure 5 IP Only Architecture Diagram



The IPX stack is loaded on systems installed with IP to give those systems the ability to execute IPX applications and to connect with IPX systems through a Migration Agent.

NetWare 6 servers installed with IP only have the following capabilities:

- They can establish NCP™ connections with clients installed with one of the install options that include IP.
- They can establish NCP connections through a Migration Agent with pre-NetWare 6 clients (these clients support only NCP connections over IPX) or with NetWare 6 clients installed with IPX.
- They can execute IPX applications and communicate directly with other NetWare 6 systems installed with IP.
- They can execute IPX applications and communicate through a Migration Agent with IPX nodes.

NetWare 6 clients installed with IP have the following capabilities:

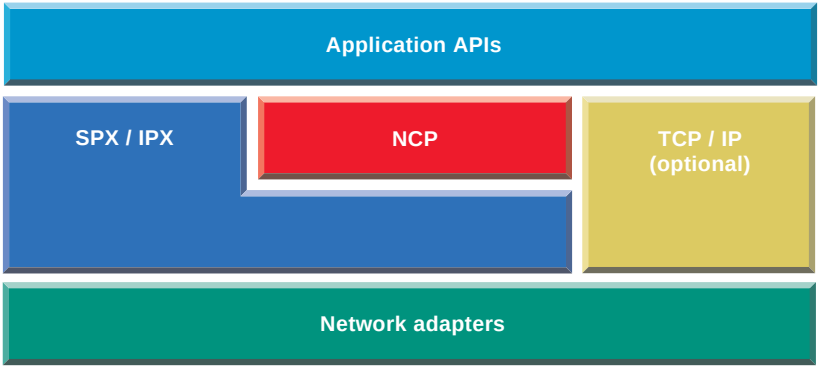
- They can establish NCP connections with servers installed with one of the install options that include IP.
- They can establish NCP connections through a Migration Agent with pre-NetWare 6 servers (these servers support only NCP connections over IPX) or with NetWare 6 servers installed with IPX only.

- They can execute IPX applications and communicate directly with other NetWare 6 systems installed with IP.
- They can execute IPX applications and communicate through a Migration Agent with IPX nodes.

IPX Install Option

These systems have the IPX stack loaded and may also have the TCP/IP stack loaded. Systems installed with IPX only are configured to establish only NCP connections over the IPX stack. Figure 6 shows the relationship between the protocol stacks when a system is installed with IPX alone.

Figure 6 IPX Only Architecture Diagram



NetWare 6 servers installed with IPX have the following capabilities:

- They can establish NCP connections with pre-NetWare 6 clients or with NetWare 6 clients installed with one of the install options that include IPX.
- They can establish NCP connections through a Migration Agent with NetWare 6 clients installed with IP.
- They can execute IPX applications and communicate directly with other IPX nodes.
- They can execute IPX applications and communicate through a Migration Agent with NetWare 6 systems installed with IP.

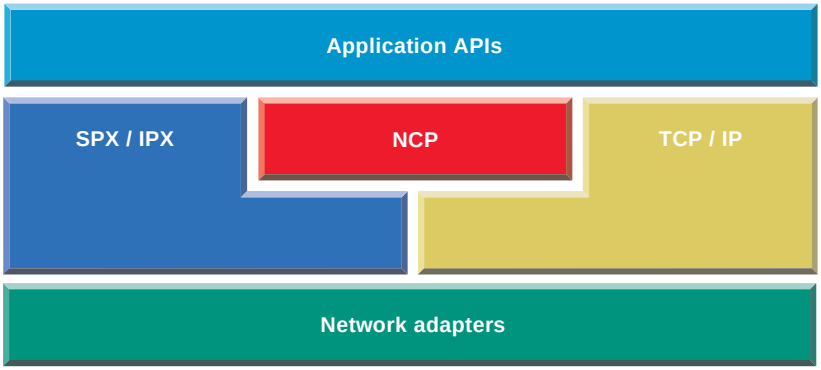
NetWare 6 clients installed with IPX have the following capabilities:

- They can establish NCP connections with pre-NetWare 6 servers or with NetWare 6 servers installed with one of the install options that include IPX.
- They can establish NCP connections through a Migration Agent with NetWare 6 servers installed with IP.
- They can execute IPX applications and communicate directly with other IPX nodes.
- They can execute IPX applications and communicate through a Migration Agent with NetWare 6 systems installed with IP.

IP and IPX Install Option

These systems have the TCP/IP and IPX stacks loaded. Systems installed with both IP and IPX are configured to establish NCP connections either over the TCP/IP stack or over the IPX stack. Figure 7 shows the relationship between the protocol stacks when installing a system with IP and IPX.

Figure 7 IP and IPX Architecture Diagram



NetWare servers installed with IP and IPX have the following capabilities:

- They can establish NCP connections with pre-NetWare 6 clients or with NetWare 6 clients without regard for the option used to install it.
- They can execute IPX applications and communicate directly with other IPX nodes.
- They can execute IPX applications and communicate through a Migration Agent with NetWare 6 systems installed with IP alone.

NetWare clients installed with IP and IPX have the following capabilities:

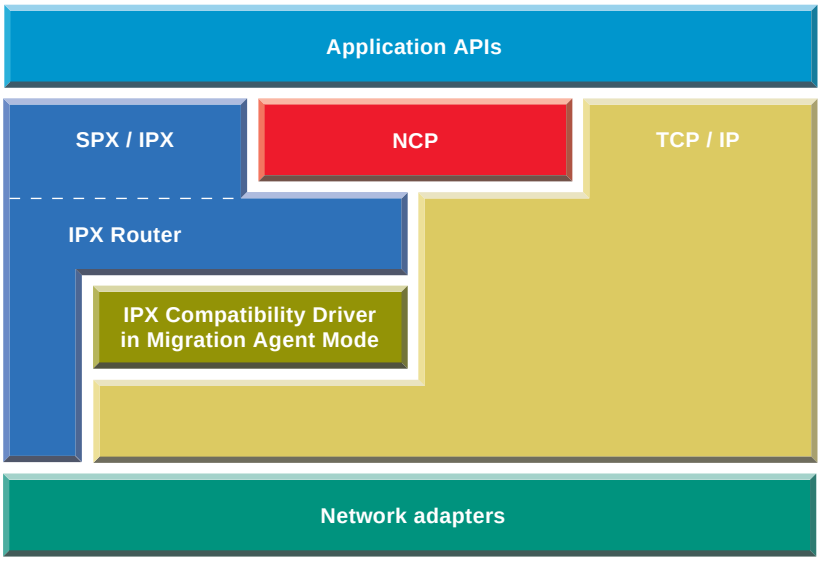
- They can establish NCP connections with pre-NetWare 6 servers or with NetWare 6 servers installed with one of the install options that include IPX.
- They can establish NCP connections with NetWare 6 servers installed with IP if the clients are able to obtain IP addresses for those servers.
- They can establish NCP connections through a Migration Agent with NetWare 6 servers installed with IP if the clients are only able to obtain IPX addresses for those servers.
- They can execute IPX applications and communicate directly with other IPX nodes.
- They can execute IPX applications and communicate through a Migration Agent with NetWare 6 systems installed with IP.

Having a NetWare client installed with IP and IPX does not guarantee that the client will be able to establish an NCP connection with a server installed with IP without the use of a Migration Agent. The type of address obtained by the client when trying to connect to a server determines the protocol stack utilized to establish the connection. Applications that obtain address information from the bindery will not be able to connect with servers installed with IP if there is no Migration Agent installed, and if the client is installed with IP and IPX. Notice that this problem does not exist if the client and the server are installed with IP alone.

Servers Installed with MA, IPX and IP

The Migration Agent can be enabled only in a NetWare 6 server installed with both IP and IPX. Figure 8 on page 40 shows the relationship between the protocol stacks when installing a system with the MA enabled.

Figure 8 **Migration Agent Architecture Diagram**



Notice that the MA makes use of the IPX router present in the IPX stack to route packets between the CMD network and the IPX networks.

NetWare 6 servers installed with the MA enabled are capable of communicating directly with other systems without regard for the install option used to install them. They are also capable of routing network traffic between IP and IPX systems.

3 Setting Up

If you want to migrate from your existing IPX-based network to a NetWare® 6 pure IP-based network, you should first read the Planning section. It discusses Compatibility Mode Drivers (CMD) and Migration Agents (MA), the building blocks needed to successfully migrate an IPX™ network to NetWare 6 and pure IP.

Also discussed in the Planning section are the NetWare 6 server and client installation options. You can install using IP only, IPX only, or IP and IPX.

The following section describes network scenarios that use the building blocks discussed in Planning. Existing networks will likely be a subset or superset of the examples presented. Regardless, once you understand how the building blocks work together, you should be able to architect your own migration strategy based on your unique network topology.

To install or upgrade a NetWare Server, see NetWare 6 Installation Guide.

To upgrade an existing server using IP only, see "Migrating IPX to IP" on page 41.

Migrating IPX to IP

There are many reasons to migrate from IPX to IP, but three of the most important are discussed in the following sections. These sections describe migration strategies that are effective in meeting the following goals.

- "Migrating to Obtain Internet Connectivity" on page 42
- "Migrating to Cut IPX Administrative Costs" on page 42
- "Migrating To Have an IP Only Network Eventually" on page 52

Migrating to Obtain Internet Connectivity

To add Internet connectivity to NetWare systems, simply upgrade to NetWare 6 using the IP and IPX option. This upgrade path requires administration of both IP and IPX networking protocols.Those who choose this migration path do not have to worry about setting up Migration Agents to maintain connectivity as they upgrade their systems.

Migrating to Cut IPX Administrative Costs

To migrate networks from IPX to IP and maximize the return on your investment, you will want to take advantage of the functionality provided by the IPX Compatibility drivers and the Migration Agents. The IPX Compatibility feature is critical in this scenario because it allows migration without losing connectivity and without having to upgrade existing applications.Administrators wanting to migrate networks using the IPX Compatibility feature must understand that the IPX Compatibility drivers are dependent upon the functions of SLP, and that there are costs associated with setting up an SLP infrastructure. Additionally, setting up an SLP infrastructure is an investment in the future because SLP is an emerging Internet standard that will be leveraged by future applications and devices.When using the IPX Compatibility feature to migrate, start the migration with the leaves of the network and finish with the backbone of the network, or vice-versa. Complex network environments are characterized by a backbone formed by a variety of systems interconnected with a combination of WAN and LAN links. The following topics describe three ways to migrate:

- “Migrating a Section of the Network” on page 42
- “Migrating Leaf Networks First” on page 44
- “Migrating the Backbone First” on page 46

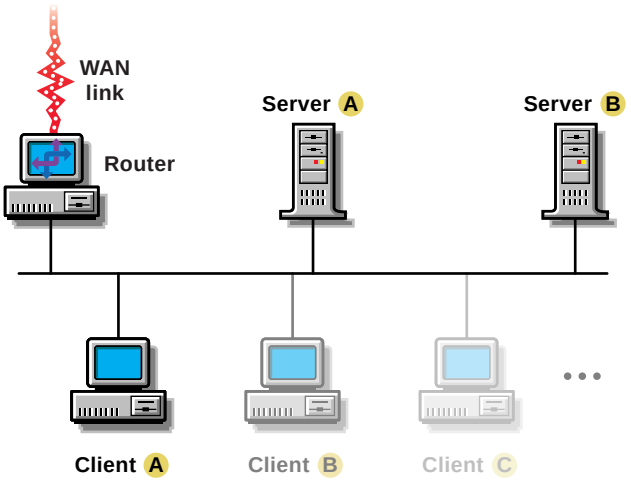
Migrating a Section of the Network

The steps below describe how to migrate a section of the network. To complete this procedure successfully, the network section being migrated must not be used to interconnect other sections of the network using IPX. The following steps allow upgrading or installing clients and servers in a phased manner without losing connectivity.

1. Select and upgrade/install some servers to serve as Migration Agents in the network section to be migrated.
2. Upgrade/Install all servers in the network section using the IP and IPX option.
3. Upgrade/Install all clients in the network section using the IP only option.
4. Modify the configuration of the servers in the network section to be IP only.
5. Turn off IPX networking between the selected section of the network and the rest of the network.

The following figure shows how the steps above are applied in a sample network.

Figure 9 Migrating a Section of the Network



1. Upgrade Server A to NetWare 6 as a Migration Agent.
2. Upgrade Server B to NetWare 6 using the IP and IPX install option.
3. Upgrade Server B to NetWare 6 using the IP and IPX install option.
4. Unbind IPX from the network adapters in server B and load scmd.nlm. Unbind IPX from the network adapters in server A and reload scmd.nlm without the Migration Agent option.
5. Turn off IPX routing at the router.

Migrating Leaf Networks First

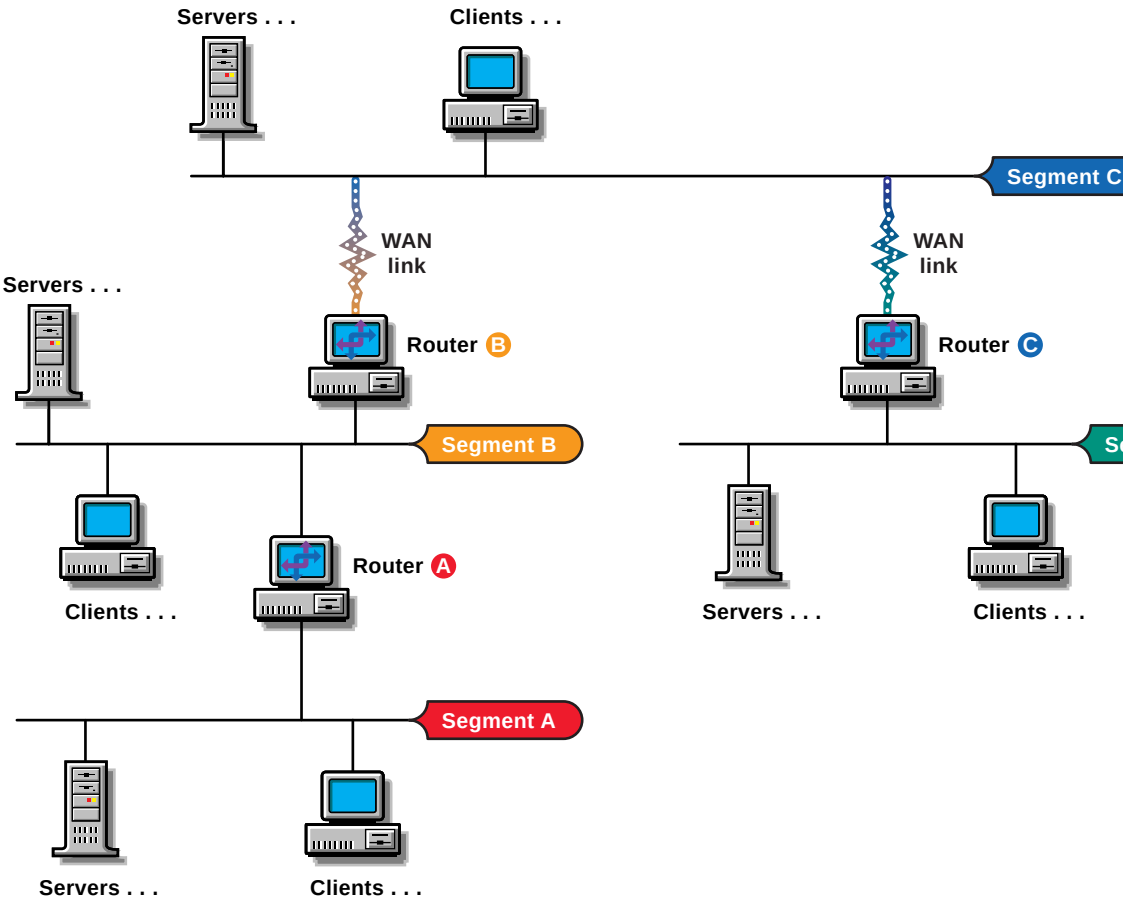
Migrating leaf networks first reduces the impact of the migration on the IPX routing infrastructure of the network, and it allows the administrator to focus efforts on specific sites. However, since the backbone is the last portion of the network migrated, administrative costs may not be offset as quickly.

The steps below describe how to migrate a network from IPX to IP starting with the leaf networks first.

1. Identify the nodes and links that form the backbone of the network.
2. Select and upgrade/install some servers in the backbone to serve as Migration Agents.
3. Select the leaf portion of the network to be migrated. This may be a group of segments connected to the backbone via a WAN link. Migrate the selected portion of the network following the steps outlined in "Migrating a Section of the Network" on page 42.
4. Repeat Step 3 until all networks connected to the backbone are migrated.
5. Migrate the backbone section using the steps outlined in "Migrating a Section of the Network" on page 42.

The following figure shows how the steps above are applied in a sample network.

Figure 10 Migrating a Leaf of the Network First



1. Identify Segment C as the backbone.
2. Upgrade/Install two servers in Segment C as NetWare 6 Migration Agents.
3. Upgrade/install servers in as NetWare 6 Migration Agents to minimize performance degradation while these segments are being migrated.
4. Migrate Segment A and Segment B, following the steps outlined in "Migrating a Section of the Network" on page 42.
5. Turn off IPX routing in routers A and B when all the nodes in the section have been migrated to IP only.
6. Migrate Segments C and D following the steps outlined in "Migrating a Section of the Network" on page 42.

Migrating the Backbone First

Migrating the backbone first alleviates administrative costs associated with maintaining IPX over the backbone. This migration path requires the following before IPX routing is disabled on the backbone:

- Migration Agents at each of the segments connected to the backbone
- Backbone Support feature enabled in the Migration Agents

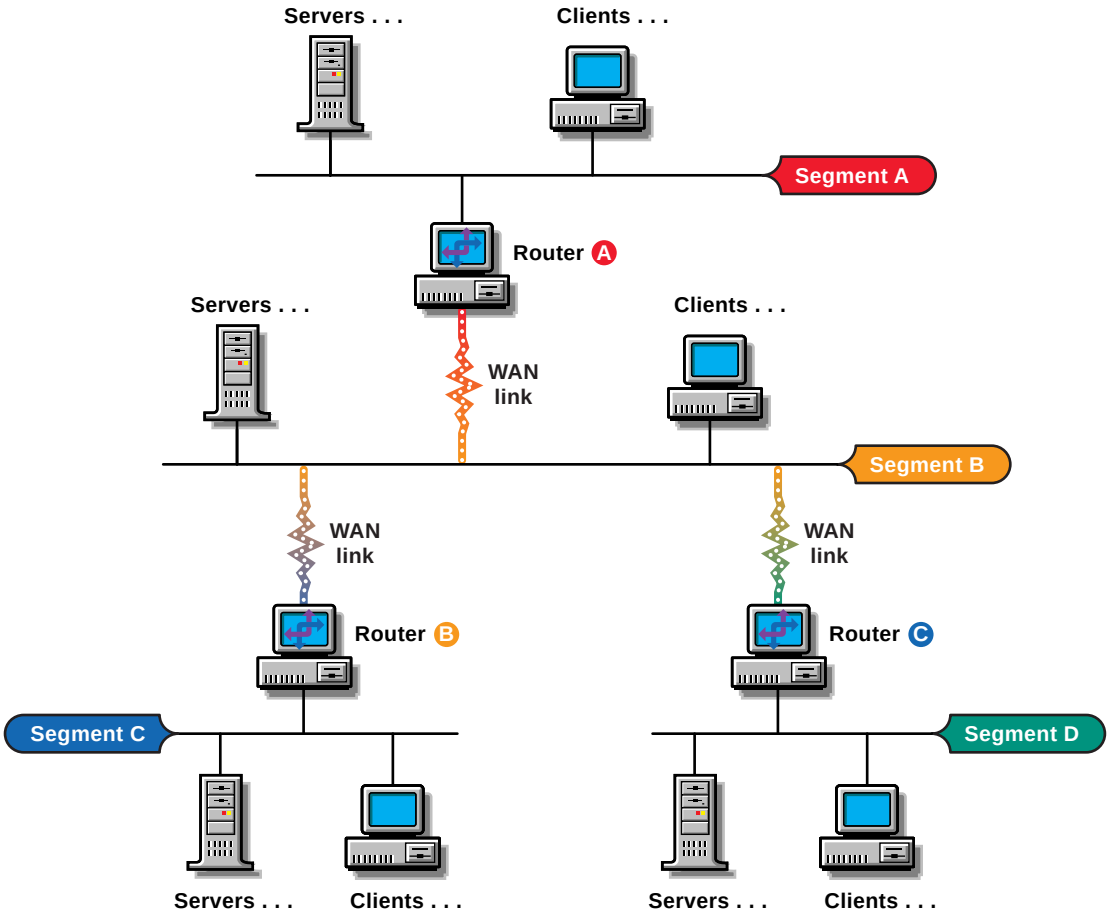
Migration Agents with the Backbone Support feature enabled can interconnect IPX segments by exchanging RIP and SAP information and by routing encapsulated IPX datagrams.

The steps below describe how to migrate a network from IPX to IP starting with the backbone first.

1. Identify the nodes and links that form the backbone of your network.
2. Select and upgrade/install some servers in each of the segments connected to the backbone to serve as Migration Agents with the Backbone Support feature enabled.
3. Migrate the backbone section using the steps outlined in “Migrating a Section of the Network” on page 42.
4. Select a leaf portion of the network to migrate. This can be a group of segments connected to the backbone via a WAN link. Migrate the selected portion of the network following the steps outlined in “Migrating a Section of the Network” on page 42.
5. Repeat Step 4 until all networks connected to the backbone are migrated.

The following figure shows how the steps above are applied in a sample network.

Figure 11 Migrating the Backbone First



1 Identify Segment B as the backbone.

2 Upgrade/Install one or two servers in segments A, C, and D as NetWare 6 Migration Agents with the Backbone Support feature enabled.Migrate Segment B (the backbone segment) using the steps outlined in "Migrating a Section of the Network" on page 42. Turn off IPX routing in routers A, B, and C to complete the migration of Segment B. Migrate segments A, C, and D using the steps outlined in "Migrating a Section of the Network" on page 42.

Avoiding Inefficient Routing

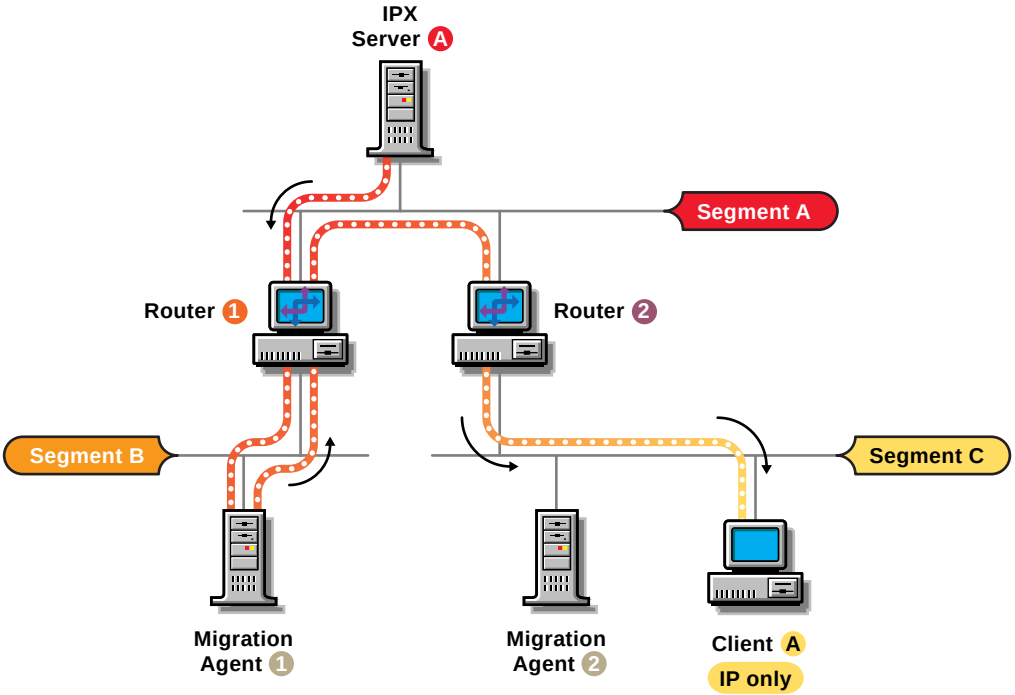
The following two examples show problems that you can avoid by carefully selecting the placement of Migration Agents in the network.

Example 1

The following figure shows a client installed as IP only in Segment C trying to communicate with an IPX server in Segment A. The IPX server knows that the client is part of the virtual CMD network and that Routers 1 and 2 present equally efficient paths to the CMD network server (the Migration Agent servers present the CMD network route to the routers attached to their network segment).

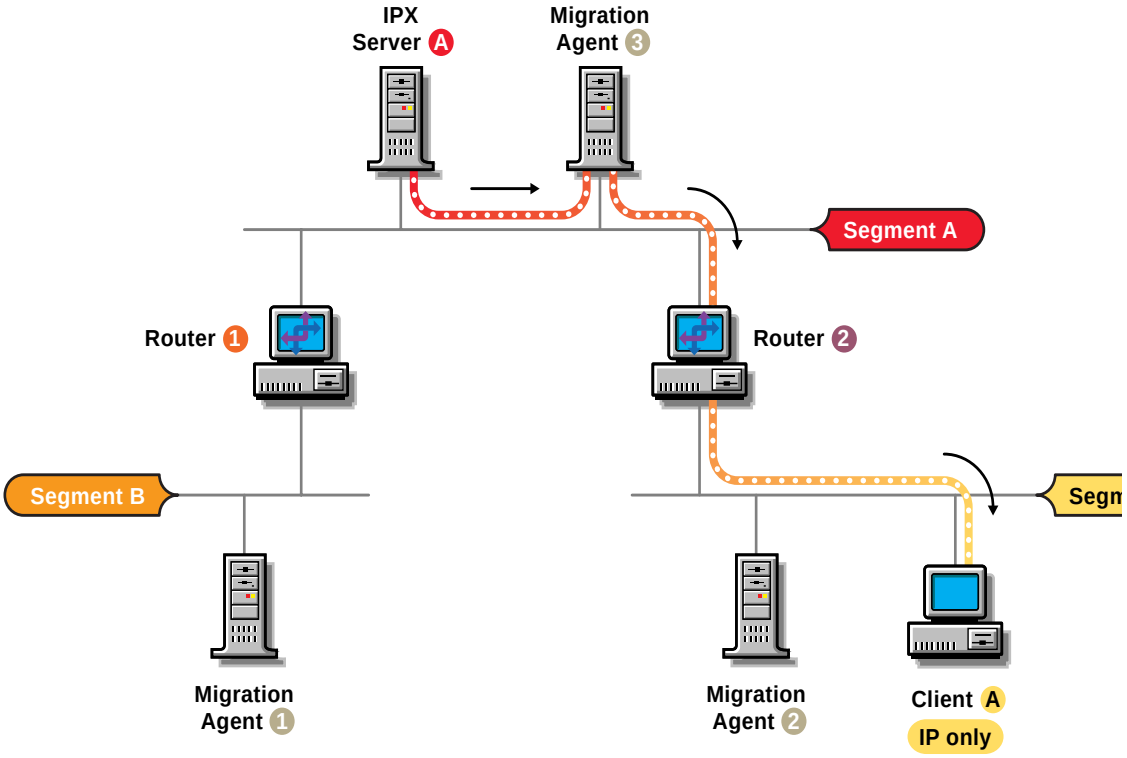
Under this scenario, Server A might choose to route packets to Client A through Router 1, resulting in the packets following the inefficient path shown by the broken line in the figure.

Figure 12 Sample Network Setup for Example 1



The problem presented here could be solved by placing a Migration Agent in Segment A as shown in the foloowing figure. The Migration Server would then present to Server A the best route to the CMD network and the packets from Server A to Client A would follow the path shown by the broken line.

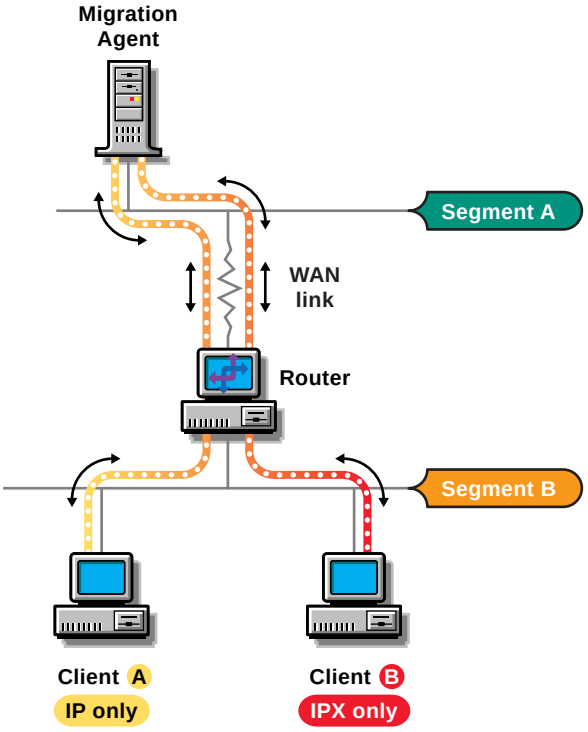
Figure 13 Sample Network Setup for Example 1



Example 2

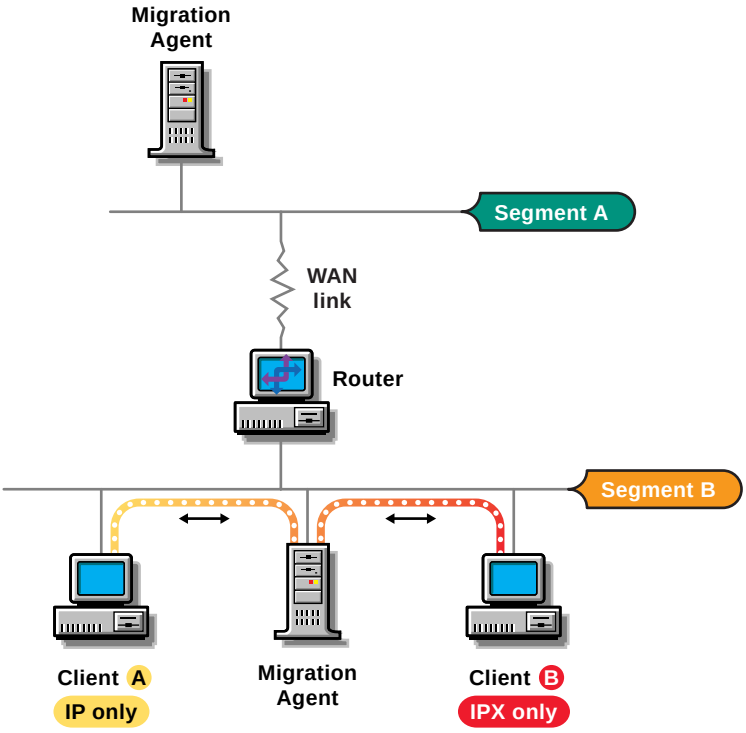
Figure 14 on page 50 shows segments A and B interconnected via a WAN link. Nodes A and B want to communicate, but they can do so only through the Migration Agent in Segment A. Under this scenario, packets sent between Node A and Node B are forced to traverse the WAN link twice, as shown by the broken line, resulting in poor performance.

Figure 14 Sample Network Setup for Example 2



The problem could be solved by placing a Migration Agent in Segment B as shown in the following figure.

Figure 15 Sample Network Setup for Example 2



SAP/RIP Filters and the Migration Agent Backbone Support Feature

If the Backbone Support feature of the Migration Agents is enabled, then the SAP/RIP information exchange between these agents can bypass the SAP/RIP filters that you might have set up in your routers. Refer to the Migration Agent documentation to learn how to set up SAP/RIP filters using the Migration Agents.

Placing of SLP Directory Agents

If you set up the SLP infrastructure using Directory agents, and if you rely on the IPX Compatibility feature to accomplish the migration, you must place Directory agents so as to minimize the round trip distance between the IP only nodes and their closest Directory agent. This is necessary to avoid having IPX applications timing out when they perform RIP or SAP requests.

Turning Off Microsoft IPX Networking

Clients might be set up to use Microsoft Networking over IPX and/or IP. If clients are set up this way and you want to migrate them from IPX to IP, you should first enable Microsoft Networking over TCP/IP and then disable Microsoft Networking over IPX. This might be necessary to reduce the demand on the services provided by the IPX Compatibility feature.

Migrating To Have an IP Only Network Eventually

Use this migration method when pure IP is desired but there is no immediate need to remove IPX from the network.This migration path requires migration of all applications from IPX to IP before IPX is disabled on the network.Applications are considered IPX applications if they use the interfaces provided by the IPX stack, or if they specify IPX addresses when trying to establish NCP connections. The best way to identify IPX applications is to run them on a test network on which IPX is absent (no IPX stacks loaded). Many applications let you specify the networking protocol to use when communicating.NetWare clients must be configured twice during the course of the migration. The cost of modifying client configurations can be minimized by taking advantage of the Automatic Client Upgrade feature for Novell Clients and the Workstation Manager feature of NDS®.If you later discover that applications require IPX, you must switch to one of the other migration strategies:

- Migrating to Obtain Internet Connectivity (page 42)
- Migrating to Cut IPX Administrative Costs (page 42)

Migrating from IPX to IP without Using the IPX Compatibility Feature

The steps below describe how to migrate a network from IPX to IP without relying on the IPX Compatibility feature.

1. Identify IPX applications and make sure that they can be configured/upgraded/replaced to run over the TCP/IP stack.
2. Start upgrading/installing your servers and clients using the IP and IPX option.
3. Start migrating applications from IPX to IP.

4. Turn off IPX networking at the routers when all the IPX applications have been migrated and all the NetWare servers and clients have been upgraded/installed using the IP and IPX option.
5. Modify the configuration of the NetWare servers and clients to be IP only servers and clients.

Configuring the Compatibility Mode

Compatibility Mode can be loaded in two different modes. When you enter the command, SCMD.NLM, the product is loaded in Compatibility Mode Server mode. To enable the Migration Agent (MA) use the /MA option.

Enabling the Migration Agent

By default, loading SCMD.NLM makes a server a simple Compatibility Mode server. To force it to act as a Migration Agent, enter the following command:

```
Load Scmd.Nlm /ma
```

The Compatibility Mode will act as a Migration Agent which can communicate and exchange details about connected Internetwork Packet Exchange (IPX) service information with similar Migration Agents. This facilitates connecting disconnected IPX segments across an IP backbone.

Changing the CMD Network Number

This option can be used when the SCMD.NLM is running either in the Compatibility Mode Server or Migration Agent mode.

By default, the CMD IPX network number is set to FFFFFFFD. This can either be changed through the Monitor screen or using the SET command line parameter.

Set Cmd Network Number=*XXXXXXX*

The SCMD module must be loaded before changing the value. Subsequently, unload and reload the module for the change to take effect permanently.

Optionally, the CMD network number can be changed dynamically while loading the module. To do this, at the console prompt type:

Load Scmd /Net=*XXXXXXX*

IMPORTANT: In NetWare 4.11 this option changes the network number temporarily. Once you unload and reload without specifying the /Net option, it will reset to the original network number.

Setting the Preferred IP Address

If multiple IP interfaces are present in the server, you can set the preferred IP address to be used by CMD. Enter the following command:

Set Preferred IP Address=*XX.XX.XX.XX*

This option can be used when the SCMD.NLM is running either in the Compatibility Mode Server mode or Migration Agent mode.

Optionally, the Preferred IP address can be changed dynamically while loading the module. To do so, at the console prompt type:

Load Scmd /PrefIP=*XX.XX.XX.XX*

Configuring the Preferred Migration Agent

The CMD server when configured on the network, will register itself with Service Location Protocol (SLP) and register information about the Migration Agents. It will query the SLP Server Agent or Directory Agent every five minutes to refresh its records. CMD clients attached to this server can access all the services that the server can access. This option can be used only when SCMD.NLM is running in CMD Server mode.

If there are any Migration Agents available on the network, the CMD server will discover the registered Migration Agent from the SLP database and register it with the Migration Agent.

IPX services discovered by the Migration Agent are not registered with the SLP database. The Migration Agent will initially register its own services with SLP and get information about all the registered services. Once the CMD server has registered with the Migration Agent, it will get updates from the Migration Agent through a transfer.

The CMD server algorithm will discover the best Migration Agent registered with SLP. However, you can statically set the list of Migration Agents by typing:

Set Preferred Migration Agents List = IP Address/

You can specify a list of preferred migration agents which this node will be using. The list should not exceed five, should be separated by semi-colons, and end with a slash (/). For the changes to be effective, unload and reload the SCMD.NLM.

Setting the SCMD.NLM to Provide IP Backbone Support

By default, IP Backbone is enabled when the SCMD.NLM is loaded using the /MA option. Ensure the following:

- Each IPX disconnected network has at least one Migration Agent running SCMD.
- All the Migration Agents should have NetWare Link Service Protocol (NLSP) routing enabled and should have the same CMD network number.
- SLP visibility exists among all the Migration Agents.
- User Datagram Protocol (UDP) communication (Port 2645) is enabled between all the Migration Agents.

To check whether IP Backbone is working, enter the following command:

```
Display Servers
```

This command will display all the services of which the server is aware.

Configuring for SLP Independent Backbone Support

You can configure CMD not to be dependent on SLP to discover other Migration Agents in the network. By default, the product uses SLP to discover Migration Agents in the network. To make it SLP independent, enter the following command at the console prompt:

```
Set No SLP Option = ON
```

When the product is operating in the SLP independent mode, you can set the time for the Migration Agents to exchange discovery information with each other. The default value is 10 minutes.

NOTE: The time set using this option will not affect the actual service and route information exchange.

To set the communication time, enter the following command:

```
Set MA Communication Time = X minutes
```

The communication time should be the same for all the Migration Agents on the network.

Optionally, you can load CMD in the SLP-independent mode and set the communication time for the Migration Agents to communicate with each other. To do so, at the console prompt type:

Load Scmd /Noslp /Synctime = *X minutes*

You can specify the Preferred Migration Agents which this node will be using. To set the Preferred Migration Agent List, enter the following command:

Set Migration Agent List = IP Address/

This list should not exceed five, the IP addresses should be separated by a semi-colon, and the list should end with a slash (/). Optionally, you can load CMD specifying the preferred Migration Agents which this node will be using. To do so, enter the following command at the console prompt:

Load Scmd /ma/ noslp /MAADDR=*XX.XX.XX.XX;XX.XX.XX.XX/*

Enable Filtering

You can configure the Migration Agent to filter some of the IPX services and/or networks between two Migration Agents. This option can be used only when SCMD.NLM is running in Migration Agent mode. Before using the filtering option, run the FILTCFG utility to set the filters. Refer to the FILTCFG documentation available at http://www.novell.com/documentation/lg/nw5/docui/index.html.

To enable filtering, enter the following command:

Scmd /filter

Viewing the Migration Agent List

You can in identify the number of Migration Agents the CMD server knows at any point. To view the information, enter the following command:

Scmd /MAList

Updating the Router Table

You can update the SAP and RIP information in the Router table using the following command:

Scmd /Sync

The information will be gathered from the Migration Agents to which the CMD server is connected, and the Router table will be updated. This option can be used when the product is running in CMD server mode.

Viewing the CMD Server Statistics

You can use the statistics option to view the current status of the CMD server or Migration Agent. Enter the following command to view the information on the CMD information screen:

```
Scmd /stat
```

If you want the information to be sent to a file, enter the following command:

```
Scmd /stat [/Dump]
```

The information will be written in the CMDSTAT.DAT file in the SYS:\ETC\ directory.

You can use the /search option to list the names of the services the CMD server or the Migration Agent have located. You can also search for the net number of the CMD server or Migration Agent. Enter the following command:

`Scmd /search [NAME=`*`service name`*`] [NET=`*`net number`*`] [/Dump]`

The parameter *service name* can take one of the following values:

- The wildcard *, which will locate all the services in the network
- The exact name of the service, for example BLR-ENGR3
- The name followed by an asterisk (*), for example BLR*

For the parameter net number you can enter **FFFFFFF** to list all the services in the network, or you can enter the matching net number.

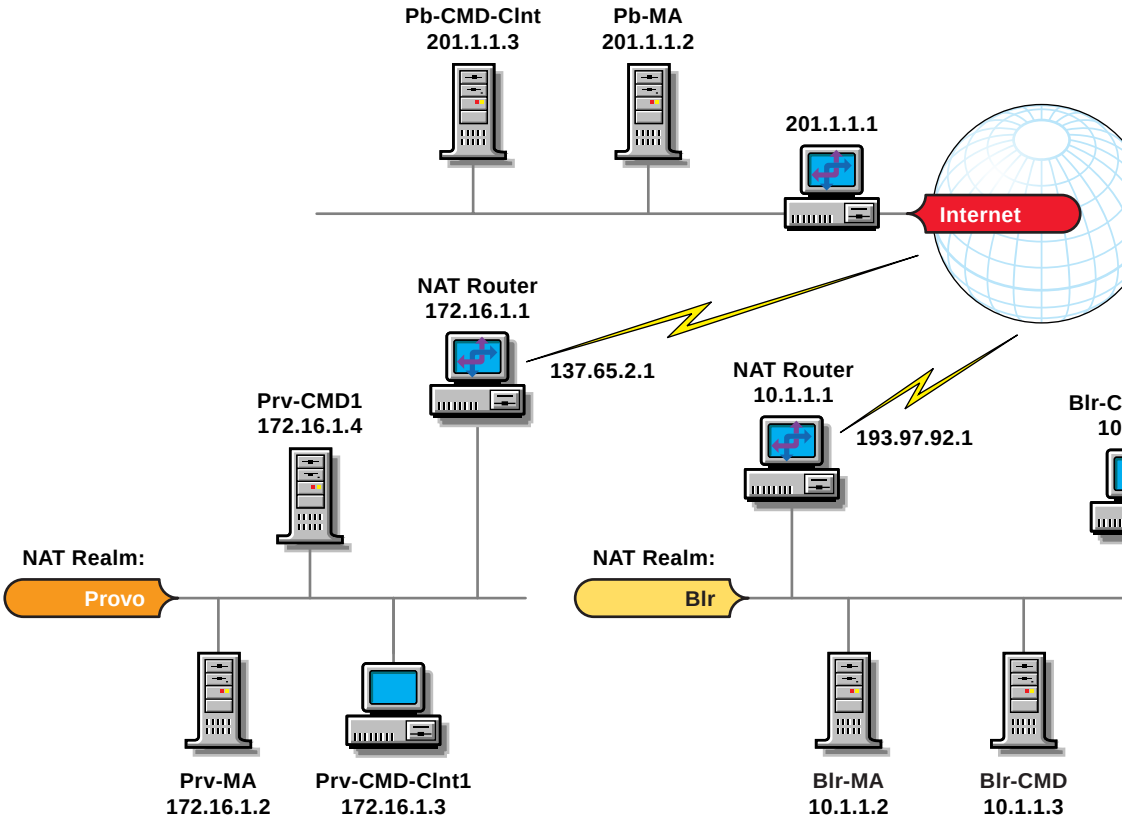
Supporting the Network Address Translator

Network Address Translation (NAT) provides a transparent routing solution using the hosts in a private network that can access an external network and vice versa. IP address translation is required when a network's internal address cannot be used outside the network either for reasons of privacy or because the IP addresses are invalid for use outside the network. Since CMD packets have IP addresses inside their payload, IP address translation cannot work over NAT without this feature.

The following diagram is an example on how NAT support can be implemented. The CMD network spreads across the intranet (which uses the private addressing scheme) and Internet (which uses the public addressing scheme). A network with NAT enabled at its border is called a private realm. The network may or may not use private IP addresses. The Internet is called a

public realm. A realm should be assigned a unique name. The next figure depicts two private routing realms, BLR and Provo, connected by a public realm to the Internet. To enable CMD communication between BLR and Provo, use the options mentioned in “Configuring CMD to Support NAT” on page 59.

Figure 16 **Two private routing realms, BLR and Provo, connected by a public realm to the Internet**



A sample configuration for a section of the network is given below:

Prv-MA	Blr-CMD	PB-CMD-CLNT
NAT realm = Provo	NAT Realm = Blr	Preferred MA List = 172.16.1.2 201.1.1.2
Public IP Address = 137.65.2.2	Public IP Address = 193.97.54.3	NA
Public IP Subnet = 255.255.0.0	Public IP Subnet = 255.255.255.0	NA
Local client number = 172.16.0.0/	Local client number = 0.0.0.0/	NA

The NAT support feature cannot be used when CMD is running in SLP-independent mode. The CMD to MA communication across NAT is not supported. We recommend that the CMD should be loaded as an MA for such a communication to happen.

Configuring CMD to Support NAT

To enable the NAT support feature through the SET command, enter the following command:

```
Set Cmd Nat Support Option = ON
```

If SCMD.NLM is running, unload and load the NLM for the changes to be effective.

To configure CMD to support NAT, as a load line command, enter:

```
scmd /ma /nat;Public IP Address;Public IP Subnet;Nat Realm
  Name;Local IP Network no.1;...Local IP Network no.n /
```

The load line command is equivalent to using the SET command options given below. The parameters are explained in “NAT Support Configuration Parameters” on page 60.

If the MA is in a public realm, enter the following command to support NAT:

```
scmd /ma /nat
```

Here, parameters like Public Address, Private IP Subnet, etc., will be assigned default values.

You can also use the SET command to configure the CMD for NAT support. These options are listed below. To assign the public IP address, enter the following command:

Set Public IP Address = *XX.XX.XX.XX*

To assign the public IP subnet, enter the following command:

Set Public IP Subnet = *XX.XX.XX.XX*

To assign the NAT realm, enter the following command:

Set NAT Realm Name = "string"

To assign the local clients IP network numbers, enter the following command:

Set Local Clients IP NetNumber List =
XX.XX.XX.XX;XX.XX.XX.XX/

NAT Support Configuration Parameters

Public IP Address

This is the IP address assigned by the NAT device to a server. Enter the value in dotted decimal format. By default, the value is set to "0.0.0.0". The public address is a unique global IP address that is statically configured in the NAT router.

Public IP Subnet

This is the subnet number of the public IP address assigned by the NAT device to a server. Enter the value in dotted decimal format. By default, the value is set to "0.0.0.0".

NAT Realm

This is a realm identifier given to a private routing realm. Enter a string (not exceeding 30 characters). By default, the value is set to "NONE". The name should be unique for all private realms.

Local IP Network no.1...Local IP Network no.n

These are a list of IP clients network numbers. Enter the value in dotted decimal format ; separate each network number by a semi-colon. These network numbers correspond to clients on the private realm of the CMD server. You need to specify network numbers of local CMD clients when CMD clients on the public network are statically pointng to CMD servers or MAs in the private realm to avoid any errors. If no CMD clients in the public network are pointing to CMD servers or MAs in the private realm, it is not necessary to set this parameter.

Troubleshooting the NAT Support Feature

If the NAT support feature is not working, check whether:

- The Service Location Protocol Directory Agent (SLPDA) has been loaded on one server and pointed to by all the servers that need to communicate using NAT.
- The NAT router public interface has been set to RIP Receive Only.
- All clients in the private realm are pointing to a private interface of the NAT router.
- Default routers are specified on servers.

Set the value for NAT Dynamic mode to pass through to "ON". Also, use display SLP services to see if the services are visible on both sides of the NAT router.

4 Optimizing

Once your network is running, the protocols associated with IP and IPX™ are largely responsible for auto-tuning themselves based on network conditions. There are, however, some settings that you can change to further optimize the way your server receives and forwards packets. See the following topics:

Using Large Internet Packets

Large Internet Packet (LIP) functionality allows the maximum size of internetwork packets to be increased. (Formerly, the maximum size was 576 bytes.)

In NetWare® versions earlier than 4.11, the workstation initiated a negotiation with the NetWare server to determine an acceptable packet size. If, during this negotiation, the server detected a router between it and the station, the server limited the maximum packet size to 576 bytes.

However, some network architecture, such as Ethernet and Token ring, can support packets larger than 576 bytes. Thus LIP allows the workstation to determine the packet size based on the maximum size supported by the router.

To implement LIP functionality for a Windows 95/98 or Windows NT workstation, do the following:

1. Click Start > Settings > Control Panel > Network > NetWare Client > Properties.
2. On the Advanced Settings tab, select Large Internet Packets and click ON.

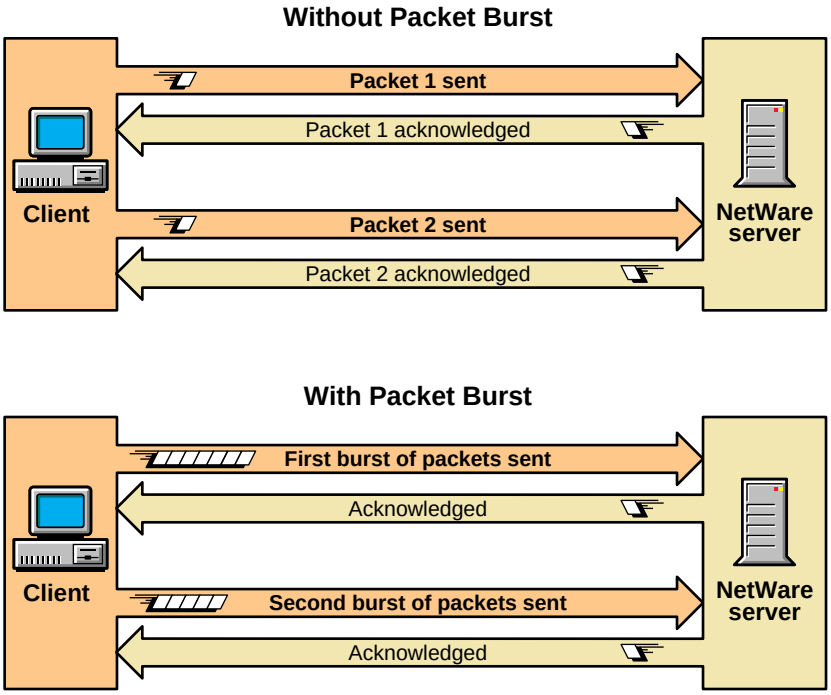
3 Click Large Internet Packet Start Size and enter the size.

4 Click OK.

Using Packet Burst

The Packet Burst™ protocol speeds the transfer of NCP™ data between a workstation and a NetWare server by eliminating the need to sequence and acknowledge each packet. With Packet Burst protocol, the server or workstation can send a whole set (burst) of packets before it requires an acknowledgment.

Figure 17 Packet Burst Protocol



By allowing multiple packets to be acknowledged, Packet Burst protocol reduces network traffic.

Packet Burst protocol also monitors dropped packets and retransmits only the missing packets.

NOTE: NetWare doesn't require an NLM™ to enable Packet Burst at the server.

For workstations to send and receive Packet Burst data, you must enable Packet Burst under the NetWare DOS Requester (for DOS or Windows 3.x) or under the Novell Client Properties, Advanced Settings (for Windows 95 or Windows NT).

For the procedures, see the help files associated with your client software.

When Packet Burst-enabled servers or workstations transfer data to servers or workstations that don't have Packet Burst enabled, the data defaults to normal NCP mode (one-request/one-response).

Increasing Maximum and Minimum Packet Receive Buffers

Packet receive buffers (also called communication buffers) store incoming data packets until they can be processed by the server.

The operating system allocates a minimum number of packet receive buffers as soon as the server boots. The minimum number is specified by the Minimum Packet Receive Buffers server parameter.

A maximum number of packet receive buffers is specified by the Maximum Packet Receive Buffers server parameter.

To determine how many buffers the server is currently allocating, refer to the Packet Receive Buffer value in the General Information window of MONITOR.

Increasing the Maximum Number of Packet Receive Buffers

If the server is slowing down and losing workstation connections, it might be running out of packet receive buffers. In this case, you can increase the Maximum Number of Packet Receive Buffers.

The General Information window of MONITOR displays the total number of packet receive buffers that are currently allocated.

1 From MONITOR's Available Options, select Server Parameters > Communications.

A list of Communications Parameters is displayed in the upper window. The scroll thumb on the right of the window indicates that you can use the arrow keys to scroll the list.

NOTE: You can also use the SET command to set comunications parameters at the server console prompt. See Reference > Utilities Reference > Utilities > SET.

2. Scroll down the Communications Parameters list.
3. Select Maximum Packet Receive Buffers.
4. Increase the value of this parameter and press Enter.

   A good guideline is to set this value to twice the size of the Minimum Packet Receive Buffer value. The changed value is now persistent.

   For additional suggestions, see the discussion of the Maximum Packet Receive Buffer parameter in SET Communication Parameters.

Increasing the Minimum Number of Packet Receive Buffers

Use the following procedure to increase the minimum number of packet receive buffers if the allocated number is higher than 10 and the server doesn't respond immediately after starting.

1. From MONITOR's Available Options, select Server Parameters > Communications.

   A list of Communications Parameters is displayed in the upper window. The scroll thumb on the right of the window indicates that you can use the arrow keys to scroll the list.

   NOTE: You can also use the SET command to change parameter values at the server console prompt. See Reference > Utilities Reference > Utilities > SET.
2. Scroll down the Communications Parameters list.
3. Choose Minimum Packet Receive Buffers.
4. Increase the value of this parameter.

   A good guideline is to allocate at least two packet receive buffers for each workstation connection. The changed value is now persistent.

   For additional suggestions, see the discussion of the Minimum Packet Receive Buffer parameter in SET Communication Parameters.

   NOTE: The Minimum Packet Receive Buffers value should be smaller than the Maximum Packet Receive Buffers value. If it is greater than the maximum value, the system changes the maximum value to match the minimum value.

5 Managing

Once your network is running, network communications management involves maintaining the physical connections between machines and maintaining the drivers that communicate with the network board. To use a network board, the LAN driver must be bound to the board. The following topics describe the procedures used to maintain communications between servers:

For ways to prevent physical communication problems, see "Preventing Cabling Problems" on page 74.

Overview of Loading and Binding LAN Drivers

After you add a network board to your NetWare® server, you must load the corresponding LAN driver. LAN drivers have .lan extensions.

When you load the LAN driver, you must specify one or more frame types for the driver. Loading a LAN driver establishes a network connection (if the server is physically connected to the network cabling). The frame type specifies how packets will be formatted for transmission across the network. You can load more than one frame type with each driver.

Once the LAN driver is loaded, you must bind the LAN driver to a communication protocol. Binding a LAN driver assigns a network communication protocol to the driver and the network board. Without a protocol, the LAN driver can't process packets, and workstations attached to the cabling scheme from that board can't log in.

The protocol is actually bound to a protocol ID (PID) that is part of a frame type. Because a frame type can have multiple PIDs, you can bind one LAN driver and one frame type to multiple protocols. You can also bind the same protocol to more than one LAN driver.

To load and bind LAN drivers, you can use

- NWCONFIG NLM™
- LOAD and BIND commands

  If you know the parameters required by the communication protocol, you can use the LOAD and BIND commands to load and bind LAN drivers at the server command line. For more information, see BIND and LOAD in *Utilities Reference*.

Loading and Binding LAN Drivers

When you bind the IPX™ protocol to a board, you specify the cabling scheme's IPX external network number.

The IPX external network number is a hexadecimal number. This number must be the same for all boards cabled together that use the same frame type.

The IPX external number must be different from the number used by boards of other frame types and must be different from the addresses of other cabling systems on the network. The cabling scheme's IPX external network number must also be different from the *internal* network address for any node on the network.

The following procedure explains how to use NWCONFIG to load a LAN driver and bind a protocol.

1. At the server console prompt, enter

   `[LOAD] NWCONFIG`

2. Select Driver Options > Configure Network Drivers.

3 If you want NetWare to automatically detect all the drivers that are compatible with your network boards, select Discover and Load Additional Drivers; otherwise, skip to Step 5.

NetWare automatically finds the parameter values for the driver, loads the driver with the parameters, and then discovers the IPX protocol for the frame type supported for the driver.

NetWare then displays the IPX internal and external network addresses in a confirmation box.

NOTE: In machines with PCI buses and sometimes in machines with EISA hardware, NWCONFIG might not detect all the drivers associated with the LAN adapters. In other cases, NWCONFIG might find more than one compatible driver for an adapter. When this happens, NWCONFIG displays a message listing the hardware. The message prompts you to press Enter to see a list of all available drivers for the hardware. Press Enter, then Go to Step 6.

4 Confirm the addresses.

After you confirm the addresses, NetWare binds the protocol to the driver.

If you want to load additional drivers that are not autodetected, continue with Step 5.

5 To select a driver from all the available drivers, choose Select an Additional Driver.

The screen displays a list of all available drivers.

6 Select the driver you want to load, or, if the driver is not on the list, press <Insert>.

If you press <Insert> to load an unlisted driver, follow the screen prompts. If you select a listed driver, continue with Step 7.

NOTE: For some drivers, a message might appear indicating that the driver must be loaded manually (at the console prompt). To load a driver manually, follow the screen prompts or press <F1> for more information.

7 Choose Select/Modify Driver Parameters.

The screen displays a window where you can set values for driver parameters. Depending on the driver you selected, a window containing protocol options might also be displayed. The cursor is active in the protocol options window, if it is displayed.

8 If the window containing protocol choices is displayed, check the protocol you want to use.

9 If you checked TCP/IP, enter the IP address and the IP mask.

10 Use the down arrow key to move the cursor to the parameter window; enter parameter values as needed.

Press <F1> for help if necessary.

In some cases, the system displays a pop-up list of values for the field from which you select the desired value. In other cases, you must type in a value and press <Enter> to move to the next field.

You can also specify a specific frame type if desired. If you do not specify a particular frame type, all frame types are loaded automatically, but only those found on the network are actually bound to the driver.

11 (Optional) To specify a particular frame type for an Ethernet driver, press <F3> to display a list of frame types. Use the arrow keys to move up and down the list. Press <Enter> to select a frame type. When finished, press <Esc>.

12 When finished, press <F10> to save the values and exit the window.

The system loads the LAN driver and then displays a confirmation window containing the command line to bind the protocol with the specified frame type.

At this point, you can either confirm binding of the protocol with the specified frame type or change the frame type.

13 To bind the protocol, press <Enter>. To display the command line with a different frame type, press <F3>, then press <Enter> when the desired frame type is displayed.

NWCONFIG automatically places the LOAD and BIND commands in the autoexec.ncf file.

Unbinding and Unloading LAN Drivers

To remove a communication protocol from a board and driver, you can use the UNBIND console command. If you have loaded the driver more than once, specify the board you want to unbind. See UNBIND in *Utilities Reference*.

When the protocol is unbound, users attached to the cabling scheme of the board can't log in. If users are already logged in, they receive a message when they attempt to access the server.

To unload a LAN driver, use the UNLOAD command or NWCONFIG. See NWCONFIGUNLOAD in *Utilities Reference*.

Using Logical Boards

A LAN driver can be loaded with multiple frame types. Each instance of a LAN driver and an associated frame type is one *logical board*. Therefore, while there might be only one physical network board in the server with one LAN driver, there can be multiple logical boards.

For example, if your server contains an NE2000™ board, you can load the NE2000 LAN driver with frame types Ethernet_802.2 and Ethernet_II. In this situation there is one physical board and one LAN driver, but there are two logical boards.

In older versions of NetWare, you could not unload individual logical boards. To remove a particular logical board, you had to unload the LAN driver, which in turn deactivated all network adapters associated with the LAN driver, and also unloaded all the frame types associated with the driver. You then had to reload the driver for each board you wanted to use and each frame type you wanted to keep. On large networks this process was extremely time consuming.

In NetWare, you can now unload, shut down, and reset individual logical boards, and also remove or reset individual adapters associated with a LAN driver.

Unloading Logical Boards

1. Determine the logical board number or name.

   A name can be assigned to a logical board when the board is loaded with the LOAD command. If no name was assigned to the board, you can determine the logical board number by using MONITOR.

   1a. At the server console prompt, load MONITOR.

   1b. Select LAN/WAN Drivers and highlight the desired LAN driver.

   A screen is displayed containing the logical board numbers and other data associated with the driver. Note the logical board number you want to unload.

2. Enter the following at the server console prompt:

   REMOVE NETWORK INTERFACE *board_number* | *board_name*

   Specify either the logical board number or the board name. NetWare unloads the logical board and deletes its resources.

Shutting Down and Resetting Logical Boards

You can shut down a logical board without removing its resources. In this case, you can restart the board, if needed, without reloading and binding the LAN driver.

1. Determine the logical board number or name.

   A name can be assigned to a logical board when the board is loaded using the LOAD command. If no name was assigned to the board, you can determine the logical board number by using MONITOR.

   1a. At the server console prompt, load MONITOR.

   1b. Select LAN/WAN Drivers and highlight the desired LAN driver.

   A screen is displayed containing logical board numbers and other data associated with the driver. Note the logical board number you want.

2. Enter the following at the server console prompt:

   SHUTDOWN NETWORK INTERFACE *board_number* | *board_name*

   Specify either the logical board number or the board name.

3. To restart the logical board, enter the following at the server console prompt:

   RESET NETWORK INTERFACE *board_number* | *board_name*

   NOTE: Resetting the logical board does not reset the network board.

Removing Network Boards

If you want to remove a network board and there is only one instance of the board in the server, you can simply unload the LAN driver and physically remove the board. Unloading the LAN driver releases the memory resources used by the board and driver. See UNLOAD in *Utilities Reference*.

However, if you have several boards of the same kind in the server, and you want to remove just one, removing the LAN driver would disable all the boards. You would then have to reload and bind the LAN driver for each board that remained in the server.

Use the following procedure to unload one board while keeping other boards of the same type enabled.

1. Determine the filename and the instance number for the board you want to remove.

   The filename is the name of the LAN driver, such as ne2000.lan.

   The board instance number is the number of the board if there is more than one board of the same type installed in the server.

   1a. At the server console prompt, load MONITOR.

   1b. Select LAN/WAN Drivers and highlight the desired LAN driver.

   A screen displays the instance number and other data for boards associated with the driver. Note the instance number for the board you want to remove.

2. Enter the following at the server console prompt:

   ```
   REMOVE NETWORK ADAPTER filename,
     [board_instance_number]
   ```

   The network driver and its resources are deleted.

Resetting Network Boards

WARNING: Resetting a network board stops whatever work the board is doing and resets it to a clean state.

Network boards will reset themselves automatically if something goes wrong. About one reset a day is normal. A great number of resets, such as one reset a minute, usually indicates a hardware problem.

Resets are included in the LAN statistics displayed in MONITOR.

Sometimes it is useful to reset a board manually if you suspect a problem with the hardware. Resetting the network board also resets the logical boards associated with the network board. (But resetting the logical board does not reset the network board). Use the following procedure to reset a board.

1. Determine the filename and the instance number for the board you want to reset.

   The filename is the name of the LAN driver, such as ne2000.lan.

   The board instance number is the number of the board if there is more than one board of the same type installed in the server. If there is only one instance of the board, you do not need the board instance number.

   1a. At the server console prompt, load MONITOR.

   1b. Select LAN/WAN Drivers and highlight the desired LAN driver.

A screen displays the instance numbers and other data for boards associated with the driver. Note the instance number for the board you want to remove.

2 Enter the following at the server console prompt:

```
RESET NETWORK ADAPTER filename,
  [board_instance_number]
```

Include the board instance number only if there are multiple instances of the same adapter in the server.

Preventing Cabling Problems

- Use the proper cabling for your network topology as specified by IEEE. Make sure cable segments do not exceed the recommended lengths.
- Make sure cable segments are properly terminated for the type of cabling being used.
- Make sure terminators and in-line cable connectors are working properly.

  If you are not sure whether a terminator or connector is working properly, replace it. If the new components work properly, discard the old ones.
- Make sure there are no breaks in the cable or shield. Use a time delay reflectometer (TDR), a LANalyzer, or a volt ohm meter (VOM) to test cabling for breaks in the cable conductor or shield.
- Make sure cabling is routed away from devices that produce high electric or magnetic fields, such as fluorescent lights, microwaves, radar, X-rays, copy machines, etc.