Chapter 21

AppleTalk

Introduction

This chapter describes the AppleTalk routing protocol, support for AppleTalk on the router, and how to configure and operate the router to act as a wide area AppleTalk router.

The AppleTalk network architecture was developed by Apple Computer Inc. to provide internetworking of Macintosh computers and other peripheral devices using LocalTalk media, and to allow seamless access to network services such as file servers and printers from the familiar Macintosh desktop environment. The open nature of the architecture has enabled the AppleTalk network system to be extended to support other media types (e.g. EtherTalk for Ethernet media), and a mixture of both Apple and non-Apple network devices on the same AppleTalk network.

AppleTalk Protocol Architecture

The AppleTalk protocol architecture is a layered protocol architecture with well defined interfaces between layers (Figure 21-1 on page 21-3). Each protocol makes use of the services provided by a lower layer protocol, to provide an enhanced service to a higher layer protocol.

The AppleTalk protocols can be considered within the framework of the OSI seven-layer Reference Model. The physical layer is represented by AppleTalk network hardware, including LocalTalk, EtherTalk and TokenTalk. The data link layer is represented by the AppleTalk link access protocols — LocalTalk Link Access Protocol (LLAP), EtherTalk Link Access Protocol (ELAP) and TokenTalk Link Access Protocol (TLAP), respectively. The network layer is represented by the Datagram Delivery Protocol (DDP). The transport layer is represented by the Routing Table Maintenance Protocol (RTMP) and the Name Binding Protocol (NBP), while the session layer is represented by the Zone Information Protocol (ZIP).

AppleTalk Nodes and Networks

An AppleTalk node is any device, such as a personal computer, file server or printer, connected to an AppleTalk network (LAN).

An AppleTalk network can be set up using a range of different media types including LocalTalk™, EtherTalk® and TokenTalk. Different physical networks can be interconnected via routers to create arbitrarily large AppleTalk internets.

LocalTalk

LocalTalk hardware is built into every Macintosh computer, Apple computer, LaserWriter printer, and many other peripheral devices. LocalTalk networks use 230.4 Kbps twisted pair cable in a bus topology. A single AppleTalk network can span up to 300 metres and support a maximum of 32 devices. The data link layer protocol used to deliver data packets between nodes on a LocalTalk network is the *LocalTalk Link Access Protocol (LLAP)*. LLAP provides a "best effort" delivery service of error free packets. It does not guarantee delivery, but every packet that is delivered is guaranteed to be error free.

Each node on a LocalTalk network is assigned an 8-bit node identifier number (node ID) as its data link address, using a dynamic assignment mechanism. When a new node is activated it selects a *provisional* node ID and then verifies the uniqueness of the *provisional* node ID by transmitting an *LLAP Enquiry control* packet to the provisional node address. If the provisional node ID is in use by another node, that node will respond with an *LLAP Acknowledge control* packet. The new node then selects another provisional node ID and repeats the process. If an Acknowledgement control packet is not received after a number of retransmissions of the Enquiry control packet, the provisional node ID is taken to be unique and used until the device is switched off. Node IDs are divided into two classes, user node IDs and server node IDs, to limit the number of enquiry packets required to confirm the provisional node ID (Table 21-1 on page 21-4). Each LLAP packet includes the node IDs of the source and destination nodes, used by the network hardware to ensure the packet is delivered to the correct node.

Figure 21-1: AppleTalk protocol architecture, with the elements supported by the router highlighted.

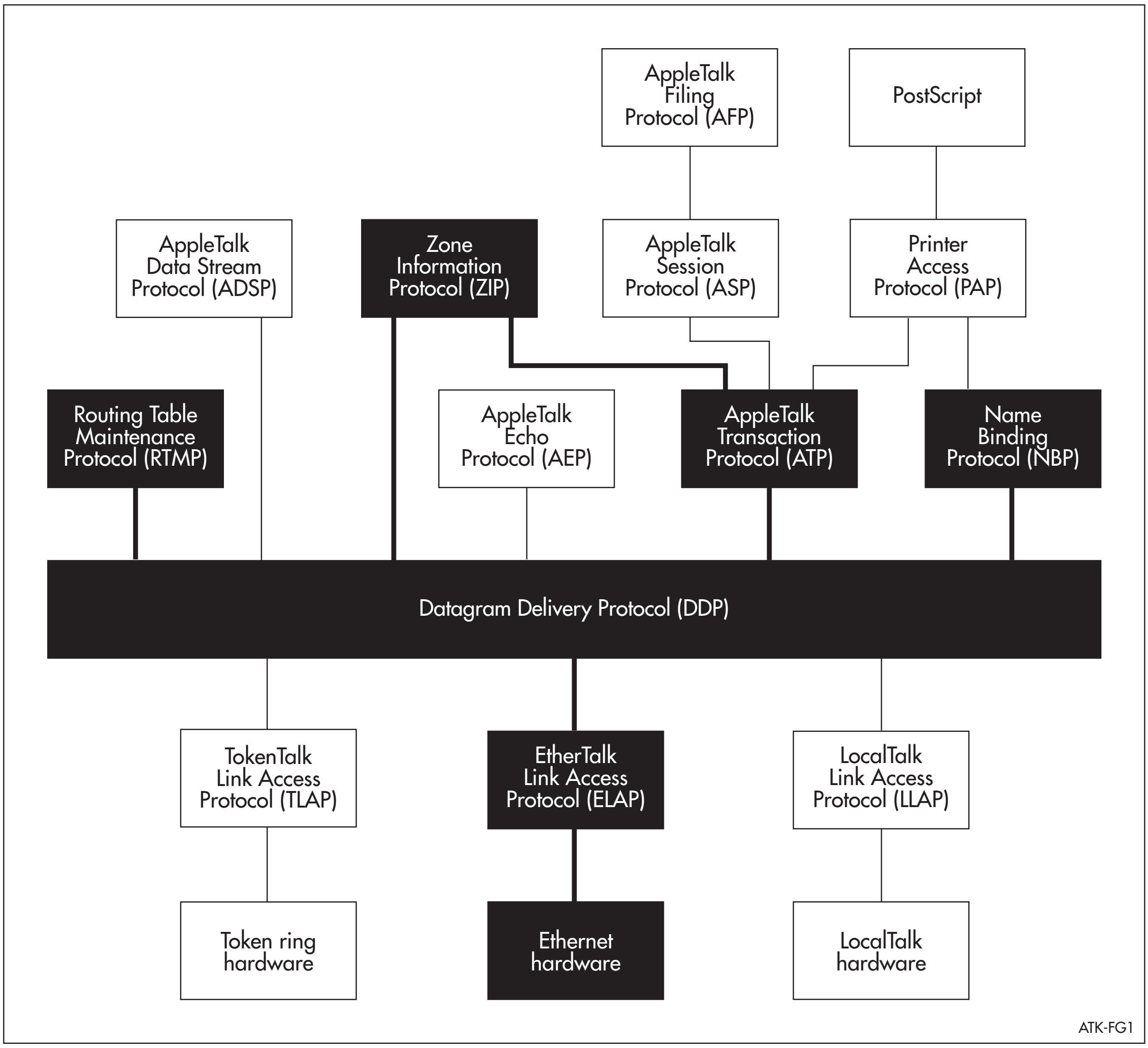

Table 21-1: Classes of LocalTalk Link Access Protocol node IDs.

Node ID Range	Description
0	Not allowed (unknown)
1-127	User node IDs
128-254	Server node IDs

EtherTalk and TokenTalk

EtherTalk provides high speed connection of devices using standard 10 Mbps Ethernet technologies. EtherTalk can support as many concurrently active AppleTalk devices as can be connected to an Ethernet network. The data link layer protocol used to deliver data packets between nodes on an EtherTalk network is the *EtherTalk Link Access Protocol (ELAP)*.

TokenTalk provides connection of devices using standard Token Ring technologies. TokenTalk can support as many concurrently active AppleTalk devices as can be connected to a Token Ring network. The data link layer protocol used to deliver data packets between nodes on an TokenTalk network is the *TokenTalk Link Access Protocol (TLAP)*.

Node addressing on EtherTalk and TokenTalk networks differs from node addressing in LocalTalk networks in that the physical hardware addressing schemes support a much larger number of devices, and therefore a physical address can not be mapped directly to the node ID of an AppleTalk protocol address. Both ELAP and TLAP rely on the *AppleTalk Address Resolution Protocol (AARP)* to provide translations between hardware addresses and AppleTalk node IDs, and for dynamic address-acquisition.

AppleTalk Address Resolution Protocol (AARP)

A *protocol stack* running within a node has its own unique *protocol address* to identify the stack among its peers on the internet and to communicate with these peer entities. A protocol address comprises a 16-bit *network number* and an 8-bit *node ID*. On a LocalTalk network, which supports a maximum of 254 devices, the AppleTalk data link address can be used directly as the lower 8 bits of the protocol address without the need for any special address resolution mechanism. On other media, such as EtherTalk, the data link address space is much larger, and can not be used directly as a portion of the protocol address. The *AppleTalk Address Resolution Protocol (AARP)* provides a mechanism for mapping protocol addresses to hardware addresses. When the protocol stack transmits a data packet it specifies the destination by its protocol address. The data link layer uses AARP to translate the protocol address into the hardware node address of the destination node.

AARP performs three related address resolution functions — translating protocol addresses into hardware addresses; dynamically determining the node's protocol address; and filtering incoming packets.

Address Translation

On each node, AARP maintains a cache of mappings between protocol and hardware addresses, called the *Address Mapping Table (AMT)*, which it uses to translate a protocol address to a hardware address. When a new address mapping is discovered it is added to the cache. When the cache becomes full, the least recently used entries are removed. Entries are aged to remove obsolete mappings — if an entry is not refreshed within a certain period it is deemed obsolete and removed.

New mappings are discovered by gleaning information from all incoming data packets intended for the node's protocol stack (which contain both the hardware and protocol address of the sender), and from responses to specific *AARP Request* packets. When a required hardware address is not found in the AMT, AARP broadcasts an *AARP Request* packet containing the protocol address for which a hardware address is required to all nodes on the network. When a node receives an AARP Request packet containing a protocol address that matches its own protocol address, it returns an *AARP Response* packet containing its own hardware address.

Dynamic Assignment of Protocol Addresses

On EtherTalk and TokenTalk networks, a protocol stack's address can be assigned dynamically by AARP. AARP selects a provisional address which is not in the AMT, and broadcasts an *AARP Probe* packet. If a node receives a Probe packet containing a provisional protocol address matching its own protocol address, it returns an *AARP Response* packet. If the probing node receives a Response packet, it selects another provisional protocol address and repeats the process. If a Response packet is not received after a specified number of retries, then AARP accepts the provisional address as the node's protocol address.

Datagram Delivery Protocol

LLAP, ELAP and TLAP provide a best-effort, node-to-node delivery of packets on a single AppleTalk network. The *Datagram Delivery Protocol (DDP)* extends the node-to-node delivery service of data link layer protocols to a process-to-process, best-effort delivery service across an internet. Processes operating within a node can attach themselves to addressable entities called *sockets*. DDP manages the exchange of packets, called *datagrams*, between any two sockets in an internet. A process attached to a socket is a *network-visible entity (NVE)*, and can be accessed from anywhere in the internet.

A socket is a logical, addressable entity within a node, identified by an 8-bit *socket number*. Sockets 0 and 255 are reserved. Sockets 1 to 127 are statically assigned sockets, reserved for clients such as AppleTalk protocols (NBP, RTMP). Sockets 128 to 254 are assigned dynamically by DDP upon request from client processes in the node. Socket numbers are unique within a node. The *internet socket address*, which combines the network number, node ID and socket number, provides an internet-wide unique address.

Each network in an AppleTalk internet is assigned a unique range of 16-bit *network numbers*. No two networks in an internet may have overlapping ranges. Network number 0 is reserved and by default refers to the local network to which the node is attached. Certain node IDs have special meaning to DDP and should not be used as part of an AppleTalk node address. Node ID 255 (0xFF), when used in conjunction with a non-zero network number, defines a network-specific broadcast directed at the specified network. Node ID 255 (0xFF), in conjunction with network number 0, defines a network-wide broadcast. Node ID 0 identifies any router on the network specified by the network number portion of the node address. Node ID 254 (0xFE) is reserved on EtherTalk and TokenTalk networks and should not be used as a node ID.

Every AppleTalk data packet includes a network number within the range of network numbers of the destination network. Routers use the destination network number in the data packet, and information from routing tables, to making routing decisions. The data packet is forwarded from router to router until it reaches the destination network where the appropriate data link protocol delivers the packet to the destination node. The *Routing Table Maintenance Protocol (RTMP)* is used by routers to create and maintain AppleTalk routing tables.

DDP is responsible for acquiring a node's AppleTalk address at startup. This address must be unique across the AppleTalk internet. On a non-extended network (e.g. LocalTalk) with only one network number, each node's 8-bit AppleTalk node ID is unique. The underlying data link layer (e.g. LLAP) is used to dynamically assign the node ID and the node's network number is then obtained from a router using an RTMP Request packet. If a router is not present on the network, the network number is set to 0.

On an extended network (e.g. EtherTalk) with a range of network numbers, nodes are differentiated by unique network number/node ID pairs, and address acquisition is a two stage process. First, a *provisional node address* is obtained through the data link layer. The node ID portion is chosen at random, and the network number portion is chosen from the startup range (0xFF00 to 0xFFFE). The network number and node ID used the last time the node was startup is used as a "hint" or "first guess". The provisional node address is used to communicate with a router on the network and discover the network number range for the network to which the node is attached. The node's actual network number and node ID are then obtained through the underlying data link layer.

Routing Table Maintenance Protocol (RTMP)

A datagram is transmitted from its source socket to its destination socket over the internet by routers. If the destination network number is on the local network, DDP uses the data link layer to deliver the datagram to its destination node. If the destination is not on the local network, DDP uses information from routing tables to forward the datagram to another router on the route to the destination network. At the destination network, the datagram is delivered to the destination node by the local data link layer.

The *Routing Table Maintenance Protocol (RTMP)* is used by routers to establish and maintain the routing tables used by DDP to forward datagrams from any source socket to any destination socket on an internet.

The hardware interfaces on a router in an AppleTalk network are referred to as ports, and are identified by a *port number*, starting with 1. A port may be connected directly to an AppleTalk network (e.g. EtherTalk), or to a wide area link. The port has an associated port descriptor which identifies the port number, the node address of the router, the network number range for the network to which the port is attached, and whether or not the port is attached to an AppleTalk network. Port numbers are assigned dynamically by the router. The port node address and port network number range are only relevant for ports connected to an AppleTalk network, and are meaningless for ports connected to a communication link.

A stable routing table contains one entry for each network that can be reached in the internet. The entry includes the port number through which packets destined for that network must be forwarded, the node address of the next router to which the packet must be sent, the distance to the destination network, and the state of the entry. The distance measure is a hop count, each hop representing a router on the path to the destination network.

The hop count field in a DDP datagram is set to 0 by the source node. Each router along the path increments the hop count by 1. The hop count is limited to 15. If a router receives a datagram with the hop count field set to 15, and the destination node is not on a network directly connected to the router, the datagram is discarded. This mechanism is used to filter out packets circulating in loops.

When a router is switched on, the routing table is initialised with entries for each AppleTalk port connected to a network with a non-zero network number range. Each router in the internet then periodically broadcasts the contents of its routing table in *RTMP Data* packets through each port. RTMP Data packets received from other routers are used to extend or update the routing table. RTMP Data packets may include routes to new networks, which are added to the routing table, or a better route to a network that is already in the routing table, in which case the routing table is updated with the new information.

To reduce RTMP Data packet size in large internets, AppleTalk uses a *split horizon* algorithm. Entries in the routing table whose forwarding port in the routing table is equal to the port out which the entry is being sent are omitted from the RTMP Data packet.

Routing table entries are aged to purge the table of obsolete or bad entries. A periodic timer, called the *validity timer*, is used to manage this process. Each entry in the routing table is assigned a state of *good*, *suspect*, *bad*, or *bad-again*. Each time the validity timer expires, all entries in the table have their state downgraded. *Good* entries are marked *suspect*, *suspect* entries are marked *bad*, etc. Entries that are already marked *bad-again* when the timer expires are deleted. The process is arrested by the reception RTMP Data packets. All entries added to the routing table or updated as the result of processing an incoming RTMP Data packet have their state set to *good*.

The propagation of this process through the internet is speeded up by RTMP using a technique called *notify neighbour*. Bad entries are identified in RTMP Data packets by a hop count of 31. When a router receives an RTMP Data packet with such an entry, the state of the entry in the routing table is automatically set to *bad*.

Network number ranges are configured as part of the port descriptors of the router ports and then propagated through the internet by RTMP. On any particular network, only one router (the *seed router*) needs to have the network number range set into its corresponding port descriptor. If there is more than one seed router on a network, they must all use the same value for the network number range.

Name Binding Protocol (NBP)

Network-visible entities (e.g. socket clients) can assign themselves one or more *entity names*. An entity name is a character string of the form `object:type@zone`. Each field may be up to 32 characters long. The `object` field describes the particular object. The `type` field specifies attributes of the object (e.g. Mailbox, Printer), and the `zone` field identifies the location of the entity by zone name. Entity names are case-insensitive. Wildcard characters can be used to match multiple names. For the `object` and `type` fields, an equal sign (=) matches all possible values. A single approximately equal sign (≈) matches zero or more characters in an `object` or `type` string. In the `zone` field, an asterisk signifies the default zone for the node specifying the name.

Names are used to identify entities because they are easier for network users to remember and they remain constant. Numeric addresses, however, change from time to time, when the device is moved or as a result of the dynamic address acquisition process at startup. For example, a portable computer will have the same name regardless of where it is connected to an internet, but its address and the routes to the entity will change as it is moved around the internet. A personal computer that remains in the same physical location will always have the same name, but its address may change each time it is switched on, as a result of the dynamic address acquisition process.

AppleTalk protocols rely on numeric addresses. Since an entity's name is constant, but its address and routes to the entity can change, a mechanism is required to dynamically translate between names and addresses. This mechanism is provided by the *Name Binding Protocol (NBP)*.

Each node maintains a *names table* of mappings between entity names and internet socket addresses for all entities in that node. NBP is used to lookup an entity's address in the name table. The NBP process uses a statically assigned socket called the *name information socket (NIS)* to accept and service requests.

Large internets can potentially present the user with very long lists of entity names. AppleTalk internets can be subdivided into AppleTalk zones, using the *Zone Information Protocol (ZIP)*. Name lookup can then be restricted to particular zones.

On a single network, name lookup is a relatively simple process. When a client requests a name lookup, the NBP process in the client firsts checks the node's own name table. If a match is not found, NBP uses DDP to broadcast an *NBP Lookup* packet over the network to the NIS. All nodes on the network with an operational NBP process receive the Lookup packet and search their own names table for a match. If a match is found a *NBP Lookup Reply* packet containing the name and internet socket address mapping is returned to the address from which the Lookup packet was received.

On an internet, the lookup process is more complicated and relies on the use of zones and the active participation of routers. When a client requests a name lookup, the NBP process in the client sends a *NBP Broadcast Request* packet to the NIS of the local router, which retransmits the lookup request as a *Forward Request* packet to the NIS in any router directly connected to each network containing nodes in the target zone of the lookup request. When a router receives a Forward Request packet it converts it to a NBP Lookup packet and broadcasts it to the NIS in all nodes in the target zone on the destination network. The NBP Lookup Reply is returned to the original requester.

Zone Information Protocol (ZIP)

A *zone* is an arbitrary subset of the AppleTalk nodes in an internet. A particular network may contain nodes belonging to any number of zones, but a particular node may belong to only one zone at a time. Zones are defined in terms of network number ranges and therefore comprise one or more entire networks. A network may be a member of more than one zone, so zones may intersect. The sum of all zones is the entire internet.

Each AppleTalk network has an associated *zones list* that specifies the zone names that may be chosen by nodes on that network during the startup process. One of the zone names in the list is chosen as the default zone.

Routers maintain a complete mapping of all zone names to their corresponding networks in the *zone information table (ZIT)*. The *Zone Information Protocol (ZIP)* manages this process. The ZIP process monitors the routing table managed by RTMP. When a new network appears, ZIP attempts to determine the network's zone list and add it to the ZIT. When a network disappears from the routing table, ZIP removes the network and its zone list from the ZIT.

Support for AppleTalk

The router supports AppleTalk Phase 2, and enables EtherTalk (Ethernet LAN) networks to be interconnected using Point-to-Point Protocol (PPP), Frame Relay and X.25 wide area links. Direct support is provided for the EtherTalk Link Access Protocol (ELAP), the Datagram Delivery Protocol (DDP), the Routing Table Maintenance Protocol (RTMP), the Zone Information Protocol (ZIP), portions of the AppleTalk Transaction Protocol required for ZIP, and the Name Binding Protocol (NBP).

An AppleTalk interface, or port, is created and associated with a physical interface (Ethernet, PPP, Frame Relay or X.25) using the command:

```
ADD APPLE PORT INTERFACE=interface [SEED=seed] [DEMAND={ON|
   OFF}]
```

The SEED parameter is only required if the interface is an Ethernet interface and the router is to act as the seed router for the AppleTalk network to which the port is attached. The value specified is the network number range for the AppleTalk network. The SEED parameter of an existing AppleTalk port can be modified using the command:

```
SET APPLE PORT=port [SEED=seed] [DEMAND={ON|OFF}]
```

An AppleTalk port can be deleted using the command:

```
DELETE APPLE PORT=port
```

If the physical interface is a Frame Relay interface or an X.25 interface, any Data Link Connections (DLCs) and MIOX circuits configured for the respective interface are available and can be used by AppleTalk. However, each DLC or MIOX circuit that is required must be explicitly added to the AppleTalk port, after the port has been added, using the respective commands:

```
ADD APPLE DLCI=dlci PORT=port
ADD APPLE CIRCUIT=circuit PORT=port
```

Individual DLCs or MIOX circuits can be deleted from the AppleTalk using the respective commands:

```
DELETE APPLE DLCI=dlci PORT=port
DELETE APPLE CIRCUIT=circuit PORT=port
```

Information about the ports, interfaces, DLCs and MIOX circuits current assigned to the AppleTalk routing module can be displayed with the commands:

```
SHOW APPLE PORT[=port]
SHOW APPLE DLCI
SHOW APPLE CIRCUIT
```

The router plays an essential role in the operation of the Zone Information Protocol (ZIP), especially in an AppleTalk internet. For this process to work, each seed router on an AppleTalk network must be configured with the list of zone names defined for the local network, using the commands:

```
ADD APPLE ZONE=zone-name PORT=port [DEFAULT]
DELETE APPLE ZONE=zone-name PORT=port
```

The list of currently defined zones can be displayed using the command:

```
SHOW APPLE ZONE
```

The AppleTalk routing module can be temporarily disabled or enabled, without affecting configuration information, using the commands:

```
DISABLE APPLE
ENABLE APPLE
```

The current state of the AppleTalk module is displayed using the command:

```
SHOW APPLE
```

The state of the AARP cache and the routing table can be displayed using the commands:

```
SHOW APPLE AARP
SHOW APPLE ROUTE
```

AppleTalk Dial-On-Demand

The router's implementation of AppleTalk supports dial-on-demand routing on AppleTalk ports associated with Point-to-Point Protocol (PPP) interfaces that use an ISDN call or a synchronous port controlling a modem as the physical interface. The PPP interface must be configured with the IDLE parameter set to a value other than OFF. See *Chapter 3, Point-to-Point Protocol (PPP)* for details of how to configure a PPP interface for dial-on-demand operation.

An AppleTalk port is configured for dial-on-demand operation with the DEMAND parameter when the port is created, using the command:

```
ADD APPLE PORT INTERFACE=interface DEMAND=ON
```

An existing AppleTalk port can be configured for dial-on-demand operation, using the command:

```
SET APPLE PORT=port DEMAND=ON
```

Routing broadcasts are suppressed for a port with the DEMAND parameter set to ON, so that only user data is transmitted or received via the port. When there has been no user data transmitted or received within the time period specified by the IDLE parameter for the PPP interface, the PPP module will automatically disconnect the WAN connection. When the AppleTalk routing module attempts to transmit data via the AppleTalk port, the PPP module will automatically reactivate the WAN connection.

Because no routing information is propagated across an on-demand link, static routes must be added to the ports at each end of the link, using the command:

```
ADD APPLE ROUTE
```

Static routes can be added to ports associated with any type of interface — Frame Relay, X.25, Point-to-Point Protocol (PPP) and Ethernet. However, adding a static route to an Ethernet port is of limited use because routes originating from routers attached to the same Ethernet segment will always be broadcast by the routers.

Static routes can be deleted using the command:

```
DELETE APPLE ROUTE
```

The command:

```
SHOW APPLE ROUTE
```

displays the contents of the AppleTalk routing table, including both static and dynamically learned routes.

Extended Ping for AppleTalk

The extended PING command supports AppleTalk networks and addressing and can be used test the connectivity between two AppleTalk nodes to determine whether or not each node can "see" the other node. *Echo Request* packets are sent to the destination address and responses are recorded. The command:

```
PING [[APPLEADDRESS=]network.node] [DELAY=seconds]
    [LENGTH=number] [NUMBER={number|CONTINUOUS}]
    [PATTERN=hexnum] [SAPPLEADDRESS=network.node]
    [SCREENOUTPUT={YES|NO}] [TIMEOUT=number]
```

initiates the transmission of ping packets. Any parameters not specified use the defaults configured with a previous invocation of the command:

```
SET PING [[APPLEADDRESS=]network.node [DELAY=seconds]]
    [LENGTH=number] [NUMBER={number|CONTINUOUS]
    [PATTERN=hexnum] [SAPPLEADDRESS=network.node]
    [SCREENOUTPUT={YES|NO}] [TIMEOUT=number]
```

As each response packet is received a message is displayed on the terminal and the details are recorded. The default configuration and summary information can be displayed with the command:

```
SHOW PING
```

The command:

```
STOP PING
```

halts a ping in progress. For a detailed description of the extended PING command, see *Chapter 8, Internet Protocol (IP)*.

Configuration Example

This example illustrates the basic steps required to configure an AppleTalk internetwork. In this example, three separate AppleTalk LANs are to be interconnected via a Point-to-Point Protocol link and a Frame Relay network (Figure 21-2 on page 21-12, Table 21-2 on page 21-13). The procedure is similar for all three routers, the main differences being the configuration of the different types of WAN link.

Figure 21-2: Example network configuration for AppleTalk routing.

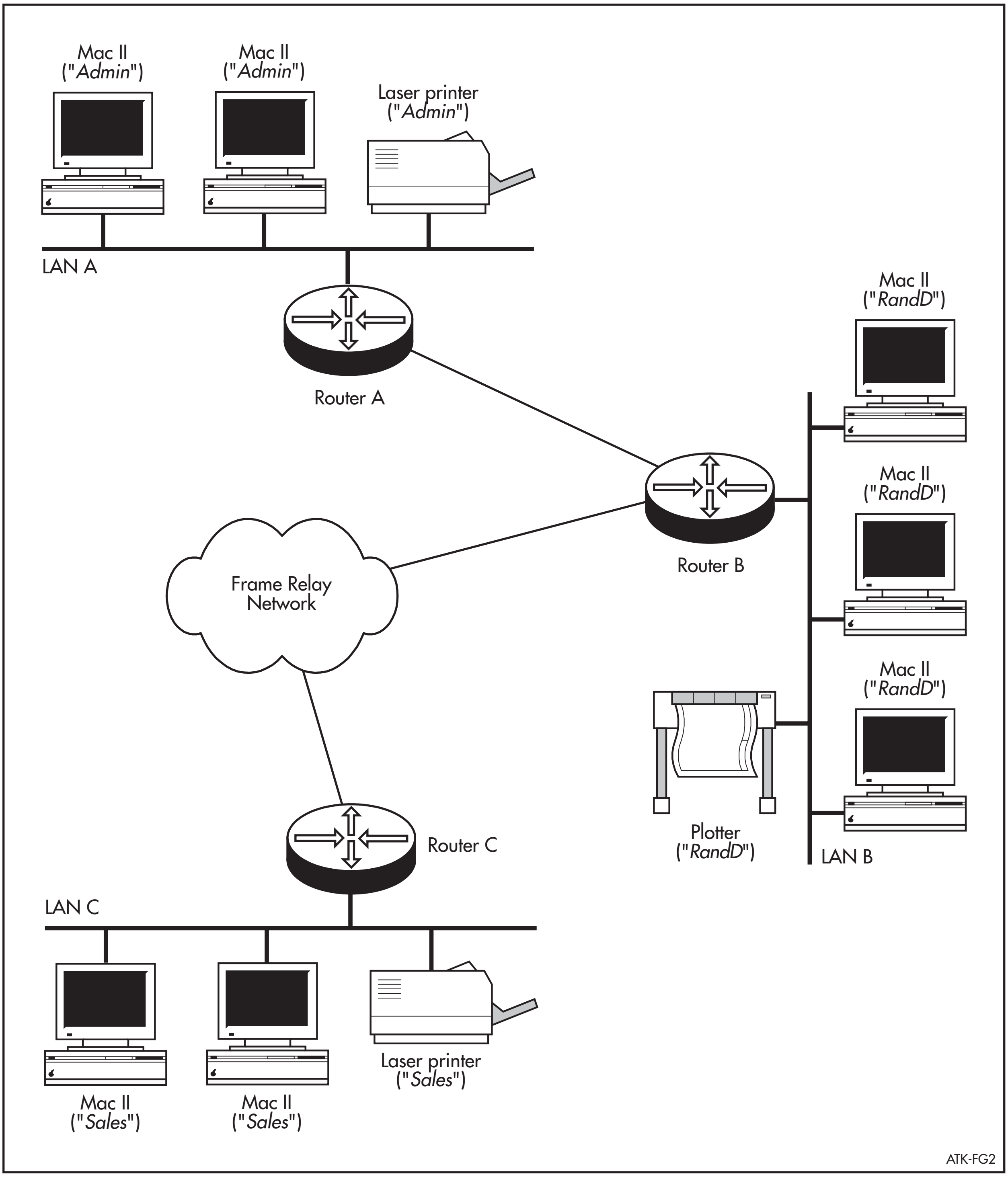

Table 21-2: Example configuration parameters for AppleTalk routing.

Parameter	Router A	Router B	Router C
Router Name	Router A	Router B	Router C
LAN Interface	eth0	eth0	eth0
LAN Name	LAN A	LAN B	LAN C
LAN Network Number Range	200–210	105–120	80–80
LAN Zone Name	Admin	RandD	Sales
WAN Interfaces	ppp0	ppp0, fr0	fr0
WAN Circuits	-	DLC 7	DLC 3

To configuring Router A:

1. **Enable the AppleTalk module.**

   Enable the AppleTalk routing module, using the command:

   ```
   ENABLE APPLE
   ```

2. **Create an AppleTalk port and an AppleTalk zone for the Ethernet interface.**

   Create an AppleTalk port on the local Ethernet interface with a network range of 200–210:

   ```
   ADD APPLE PORT INT=eth0 SEED=200-210
   ```

   Create an AppleTalk zone called "Admin" for the Ethernet network attached to the port, and make the zone the default zone for the network:

   ```
   ADD APPLE ZONE=Admin PORT=1 DEFAULT
   ```

3. **Create a PPP interface and an AppleTalk port to use the interface.**

   Create PPP interface 0 over synchronous interface 0:

   ```
   CREATE PPP=0 OVER=syn0
   ```

   Create an AppleTalk port to use PPP interface 0:

   ```
   ADD APPLE PORT INT=ppp0
   ```

4. **Check the configuration:**

   Check the AppleTalk configuration, using the commands:

   ```
   SHOW APPLE PORT
   SHOW APPLE ZONE
   SHOW APPLE ROUTE
   ```

To configuring Router B:

1. **Enable the AppleTalk module.**

   Enable the AppleTalk routing module, with the command:

   ```
   ENABLE APPLE
   ```

2. **Create an AppleTalk port and an AppleTalk zone for the Ethernet interface.**

   Create an AppleTalk port on the local Ethernet interface with a network range of 105–120:

   ```
   ADD APPLE PORT INT=eth0 SEED=105-120
   ```

Create an AppleTalk zone called "RandD" for the Ethernet network attached to the port, and make the zone the default zone for the network:

```
ADD APPLE ZONE=RandD PORT=1 DEFAULT
```

3. **Create a PPP interface and an AppleTalk port to use the interface.**

   Create PPP interface 0 over synchronous interface 0:

   ```
   CREATE PPP=0 OVER=syn0
   ```

   Create an AppleTalk port to use PPP interface 0:

   ```
   ADD APPLE PORT INT=ppp0
   ```

4. **Create a Frame Relay interface and an AppleTalk port to use the interface.**

   Create Frame Relay interface 0 over synchronous interface 1. Disable the LMI dialogue and reset the Frame Relay interface. Configure a static DLC with a DLCI of 7:

   ```
   CREATE FR=0 OVER=syn1
   SET FR=0 LMI=none
   RESET FR=0
   ADD FR=0 DLC=7
   ```

   Create an AppleTalk port to use Frame Relay interface 0, and add the static DLC:

   ```
   ADD APPLE PORT INT=FR0
   ADD APPLE DLCI=7 PORT=3
   ```

5. **Check the configuration:**

   Check the AppleTalk configuration, using the commands:

   ```
   SHOW APPLE PORT
   SHOW APPLE ZONE
   SHOW APPLE DLCI
   SHOW APPLE ROUTE
   ```

To configuring Router C:

1. **Enable the AppleTalk module.**

   Enable the AppleTalk routing module, with the command:

   ```
   ENABLE APPLE
   ```

2. **Create an AppleTalk port and an AppleTalk zone for the Ethernet interface.**

   Create an AppleTalk port on the local Ethernet interface with a network number of 80:

   ```
   ADD APPLE PORT INT=eth0 SEED=80
   ```

   Create an AppleTalk zone called "Sales" for the Ethernet network attached to the port, and make the zone the default zone for the network:

   ```
   ADD APPLE ZONE=Sales PORT=1 DEFAULT
   ```

3. **Create a Frame Relay interface and an AppleTalk port to use the interface.**

   Create Frame Relay interface 0 over synchronous interface 0. Disable the LMI dialogue and reset the Frame Relay interface. Configure a static DLC with a DLCI of 3:

   ```
   CREATE FR=0 OVER=syn0
   SET FR=0 LMI=none
   RESET FR=0
   ADD FR=0 DLC=3
   ```

Create an AppleTalk port to use Frame Relay interface 0, and add the static DLC:

```
ADD APPLE PORT INT=FR0
ADD APPLE DLCI=3 PORT=2
```

4. Check the configuration:

   Check the AppleTalk configuration, using the commands:

```
SHOW APPLE PORT
SHOW APPLE ZONE
SHOW APPLE DLCI
SHOW APPLE ROUTE
```

Command Reference

This section describes the commands available on the router to enable, configure, control and monitor the AppleTalk routing module.

See “*Conventions*” on page lxvii of *Preface* in the front of this manual for details of the conventions used to describe command syntax. See *Appendix A, Messages* for a complete list of messages and their meanings.

ADD APPLE CIRCUIT

Syntax ADD APPLE CIRCUIT=*circuit* PORT=*port*

where:

- *circuit* is a MIOX circuit name, 1 to 15 characters in length.
- *port* is an AppleTalk port number in the range 1 to 127.

Description This command adds a MIOX circuit to an AppleTalk port over an X25T interface. Up to 20 MIOX circuits may be attached to an AppleTalk port over an X25T interface.

The CIRCUIT parameter specifies the name of a MIOX circuit to be attached to the AppleTalk port. This circuit must already be configured for the X25T interface to which the AppleTalk port is attached.

The PORT parameter specifies the AppleTalk port to which the circuit is to be added. The AppleTalk port must already exist.

Examples To add the circuit “Head Office” to AppleTalk port 2, (assuming port 2 is an AppleTalk port with an X25T interface), use the command:

```
ADD APPLE CIRCUIT=”Head Office” PORT=2
```

See Also ADD APPLE PORT
DELETE APPLE CIRCUIT
SHOW APPLE CIRCUIT
SHOW APPLE PORT

ADD APPLE DLCI

Syntax

```
ADD APPLE DLCI=dlci PORT=port
```

where:

- *dlci* is the DLCI of a Frame Relay DLC, in the range 0 to 1023.
- *port* is an AppleTalk port number in the range 1 to 127.

Description

This command adds a Frame Relay DLCI to an AppleTalk port connected to a Frame Relay interface. When AppleTalk sends a broadcast to the port then each DLCI will receive a copy of the packet to be sent to the remote end.

The DLCI parameter specifies the Frame Relay DLCI to be added to the AppleTalk port. The DLCI must already exist or be configured on the Frame Relay interface to which the AppleTalk port is attached.

The PORT parameter specifies the AppleTalk port to which the DLCI is added. The AppleTalk port must already exist.

Examples

To add DLCI 23 to an AppleTalk port 1 configured to Frame Relay interface 1, use the command:

```
ADD APPLE DLCI=23 PORT=1
```

See Also

ADD APPLE PORT
DELETE APPLE DLCI
SHOW APPLE DLCI

ADD APPLE PORT

Syntax

```
ADD APPLE PORT INTERFACE=interface [DEMAND={ON|OFF|YES|
   NO}] [HINT=network:node] [SEED=seed]
```

where:

- *interface* is an interface name formed by concatenating an interface type and an interface instance (e.g. eth0). Valid interface types are ETH, PPP, FR and X25T.
- *network* is an AppleTalk network number, in the range 0 to 65279.
- *node* is decimal number in the range 0 to 127.
- *seed* is an AppleTalk network number range of the form of “nnnnnnn-nnnnnnn”. A seed specified as “aaa” is treated as “aaa-aaa”.

Description

This command adds a port to the AppleTalk routing module. A maximum of 127 AppleTalk ports can be configured at any one time.

The INTERFACE parameter specifies the lower layer interface to be used by the AppleTalk port. The interface must not already be used by another AppleTalk port. For Frame Relay and X25T interfaces, DLCIs and MIOX circuits associated with the interface must also be added, using the ADD APPLE CIRCUIT and ADD APPLE DLCI commands, before the interface can be used by the AppleTalk routing module.

The PORT parameter specifies the AppleTalk port to be added. The AppleTalk port must not already exist.

The DEMAND parameter specifies whether or not the port is connected to a dial-on-demand PPP link that will automatically be disconnected when there has been no traffic for a period set by the IDLE parameter of the associated PPP interface. AppleTalk routing broadcasts are suppressed on AppleTalk ports with the DEMAND parameter set to ON. The default is OFF.

The HINT parameter specifies the network and node to be used for the port if the network number falls within the cable range and the node number has not been used by any other device in the network. This parameter is only valid for ETH and PPP interfaces and is required for AppleTalk ports using numbered PPP interfaces. By default, AppleTalk uses addressless PPP interfaces.

The SEED parameter identifies the router as a seed router for the AppleTalk network with the specified network number. The SEED parameter is only valid for Ethernet interfaces.

The SEED and HINT parameters can be used in combination to influence the way AppleTalk assigns AppleTalk node numbers to devices on attached AppleTalk networks (Table 21-3 on page 21-17).

Table 21-3: Interaction of SEED and HINT parameters in setting AppleTalk node numbers.

SEED	HINT	Condition	Result
SEED=x-y	HINT=0:z	x <= w <= y	w:z
SEED=x-y	HINT=w:z	x <= w <= y	w:z
SEED=x-y	HINT=w:0	x <= w <= y	w:nnn
Not specified	HINT=0:z		mmm:z

If HINT is specified with a value of 0 for the network number (e.g. 0:8), then AppleTalk will attempt to select a network number (w) in the range specified by SEED with a free matching node number (8). If HINT is specified with a non-zero value for both the network and node number (e.g. 2:8) then AppleTalk will attempt to use the specified network and node number (2:8). If HINT is specified with a value of 0 for the node number (e.g. 2:0), then AppleTalk will use the specified network number (provided it is in the range specified by SEED) and select with a free node number (nnn) in that network. If SEED is not specified, HINT will not accept non-zero network numbers. In this case AppleTalk will select a network number (mmm) from the network numbers already configured on the AppleTalk network. In all cases, if the required node number is in use, the next free node number in the same network is used. If all node numbers in the network are used, AppleTalk will try other network numbers in the range specified by SEED.

Examples To add an AppleTalk port that uses Frame Relay interface 0, use the command:

```
ADD APPLE PORT INTERFACE=fr0
```

See Also ADD APPLE DLCI
ADD APPLE CIRCUIT
DELETE APPLE PORT
SHOW APPLE PORT

ADD APPLE ROUTE

Syntax

```
ADD APPLE ROUTE=network PORT=port [{NEXTHOP=network:node|
    CIRCUIT=circuit|DLCI=dlci}] [HOPS=hopcount]
```

where:

- *network* is an AppleTalk network number or an AppleTalk network number range of the form "nnnnnnn-nnnnnnn". Network numbers must decimal numbers in the range 0 to 65279.
- *port* is an AppleTalk port number in the range 1 to 127.
- *circuit* is the number of a MIOX circuit attached to an AppleTalk port over an X25T interface.
- *dlci* is the DLCI of a Frame Relay DLC attached to an AppleTalk port over a Frame Relay interface, in the range 0 to 1023.
- *node* is decimal number in the range 0 to 127.
- *hopcount* is decimal number in the range of 1 to 16.

Description

This command adds a static route to the AppleTalk routing table. The ROUTE parameter specifies the AppleTalk network number or network range for the route to be added. The route must not already exist.

The PORT parameter specifies the AppleTalk port number with which this route is associated. The port must already exist.

The NEXTHOP parameter specifies the next hop router to which packets will be forwarded. The value must be the network number and node ID of an AppleTalk device. A route must exist to the network. This parameter valid for routes associated with ports with Ethernet and PPP interfaces.

The CIRCUIT parameter specifies the number of a MIOX circuit associated with the X25T interface to which the AppleTalk port is attached. This parameter is only valid for AppleTalk ports attached to X25T interfaces.

The DLCI parameter specifies the DLCI (Data Link Connection Identifier) of a DLC (Data Link Connection) associated with the Frame Relay interface to which the AppleTalk port is attached. This parameter is only valid for AppleTalk ports attached to Frame Relay interfaces.

There is little advantage in adding a static route to an AppleTalk port that is not attached to a PPP interface, since the DEMAND option is currently only supported on PPP interfaces.

The HOPS parameter specifies this distance (number of hops) to the destination route. The default is 1.

Examples

To add a static route on port 2 with a network range of 200-210, and a distance of 2, use the command:

```
ADD APPLE ROUTE=200-210 PORT=2 HOPS=2
```

See Also

DELETE APPLE ROUTE
SHOW APPLE PORT
SHOW APPLE ROUTE

ADD APPLE ZONE

Syntax `ADD APPLE ZONE=`*`zone-name`*` PORT=`*`port`*` [DEFAULT]`

where:

- *zone-name* is a character string, 1 to 32 characters in length. Valid characters are lowercase letters (a–z), uppercase letters (A–Z) and decimal digits (0–9).
- *port* is an AppleTalk port number in the range 1 to 127.

Description This command is adds an AppleTalk zone to the zone list for a port. No more than 255 zones can be added to a particular network. If the router is a seed router for the network attached to the port, the port must have a default zone, specified by the DEFAULT parameter.

The DEFAULT parameter identifies the specified zone as the default zone for the network to which the port is connected.

The PORT parameter specifies the AppleTalk port to which the zone name is to be added. The AppleTalk port must already exist.

The ZONE parameter specifies the zone name to be added to the zone list for the port. Only seed routers need to have zones defined. Only ports connected to AppleTalk networks (i.e. Ethernet interfaces) support zone names.

Examples To add the zone name “Godzilla” to port 1 as a default zone for the attached AppleTalk network, use the command:

```
ADD APPLE ZONE=Godzilla PORT=1 DEFAULT
```

See Also ADD APPLE PORT
DELETE APPLE ZONE
SHOW APPLE ZONE

DELETE APPLE CIRCUIT

Syntax `DELETE APPLE CIRCUIT=`*`circuit`*` PORT=`*`port`*

where:

- *circuit* is a MIOX circuit name, 1 to 15 characters in length.
- *port* is an AppleTalk port number in the range 1 to 127.

Description This command deletes a MIOX circuit from an AppleTalk port. The port must be attached to an X25T interface and the specified MIOX circuit must already exist on the X25T interface.

The CIRCUIT parameter specifies the name of the MIOX circuit to be deleted from the AppleTalk port. This circuit must already be configured for the X25T interface to which the AppleTalk port is attached.

The PORT parameter specifies the AppleTalk port from which the circuit is to be deleted. The AppleTalk port must already exist.

Examples To delete circuit "Head Office" from port 1, use the command:

```
DELETE CIRCUIT="Head Office" PORT=1
```

See Also ADD APPLE CIRCUIT
ADD APPLE PORT
SHOW APPLE CIRCUIT

DELETE APPLE DLCI

Syntax DELETE APPLE DLCI=*dlci* PORT=*port*

where:

- *dlci* is the DLCI of a Frame Relay DLC, in the range 0 to 1023.
- *port* is an AppleTalk port number in the range 1 to 127.

Description This command deletes a Frame Relay DLCI from an AppleTalk port. The port must be attached to a Frame Relay interface and the specified DLCI must already exist on the Frame Relay interface.

The DLCI parameter specifies the DLCI to be deleted from the AppleTalk port. This DLCI must already be configured for the Frame Relay interface to which the AppleTalk port is attached.

The PORT parameter specifies the AppleTalk port from which the DLCI is to be deleted. The AppleTalk port must already exist.

Examples To delete DLCI 4 from the port 2, use the command:

```
DELETE APPLE DLCI=4 PORT=2
```

See Also ADD APPLE DLCI
ADD APPLE PORT
SHOW APPLE DLCI

DELETE APPLE PORT

Syntax DELETE APPLE PORT=*port*

where:

- *port* is an AppleTalk port number in the range 1 to 127.

Description This command deletes a port from the AppleTalk routing module. A maximum of 127 AppleTalk ports can be configured at any one time.

The PORT parameter specifies the AppleTalk port to be deleted. The AppleTalk port must already exist.

Examples To delete AppleTalk port 2, use the command:

```
DELETE APPLE PORT=2
```

See Also ADD APPLE PORT
SHOW APPLE PORT

DELETE APPLE ROUTE

Syntax `DELETE APPLE ROUTE=`*`network`*

where:

- *network* is an AppleTalk network number or an AppleTalk network number range of the form "nnnnnnn-nnnnnnn". Network numbers must decimal numbers in the range 0 to 65279.

Description This command deletes an AppleTalk static route. The ROUTE parameter specifies the AppleTalk network number or network range for the route to be deleted. The route must already exist.

Examples To delete the static route to the network range 200-210, use the command:

```
DELETE APPLE ROUTE=200-210
```

See Also ADD APPLE ROUTE
SHOW APPLE ROUTE

DELETE APPLE ZONE

Syntax `DELETE APPLE ZONE=`*`zone-name`*` PORT=`*`port`*

where:

- *zone-name* is a character string, 1 to 32 characters in length. Valid characters are lowercase letters (a–z), uppercase letters (A–Z) and decimal digits (0–9).
- *port* is an AppleTalk port number in the range 1 to 127.

Description This command is deletes an AppleTalk zone from the zone list for a port.

The PORT parameter specifies the AppleTalk port from which the zone name is to be deleted. The AppleTalk port must already exist.

The ZONE parameter specifies the zone name to be deleted from the zone list for the port. Only ports connected to AppleTalk networks (i.e. Ethernet interfaces) support zone names.

Examples To delete the zone "Godzilla" from port 1, use the command:

```
DELETE APPLE ZONE=Godzilla PORT=1
```

See Also ADD APPLE PORT
ADD APPLE ZONE
SHOW APPLE ZONE

DISABLE APPLE

Syntax `DISABLE APPLE`

Description This command disables AppleTalk routing in the router. This will reinitialise all data structures used by the AppleTalk routing module.

Examples To disable AppleTalk routing, use the command:

```
DISABLE APPLE
```

See Also ENABLE APPLE

DISABLE APPLE DEBUG

Syntax `DISABLE APPLE DEBUG={ALL|AARP|PACKET|PKT|ROUTE|ZIP|ZONE}`

Description This command disables AppleTalk debugging on the specified sub-protocol.

The DEBUG parameter specifies the AppleTalk protocol layer for which debugging is to be disabled. PACKET and PKT are synonyms for DDP.

Examples To disable DDP level debugging, use the command:

```
DISABLE APPLE DEBUG=PACKET
```

See Also ENABLE APPLE DEBUG

ENABLE APPLE

Syntax `ENABLE APPLE`

Description This command enables AppleTalk routing on the router.

Examples To enable AppleTalk routing, use the command:

```
ENABLE APPLE
```

See Also DISABLE APPLE

ENABLE APPLE DEBUG

Syntax `ENABLE APPLE DEBUG={ALL|AARP|PACKET|PKT|ROUTE|ZIP|ZONE}`

Description This command enables AppleTalk debugging on the specified sub-protocol.

The DEBUG parameter specifies the AppleTalk protocol layer to debug. PACKET and PKT are synonyms for DDP.

Examples To enable DDP level debugging, use the command:

```
ENABLE APPLE DEBUG=PACKET
```

See Also DISABLE APPLE DEBUG

PURGE APPLE

Syntax `PURGE APPLE`

Description This command purges all AppleTalk configuration information, but does not change the current status of the AppleTalk routing module. If the AppleTalk routing module is currently enabled, it will remain enabled after the PURGE APPLE command has been executed. Similarly, if the AppleTalk routing module is currently disabled, it will remain disabled after the PURGE APPLE command has been executed.

Examples To purge the configuration of the AppleTalk routing module, use the command:

```
PURGE APPLE
```

See Also DISABLE APPLE
ENABLE APPLE

RESET APPLE

Syntax `RESET APPLE`

Description This command resets the AppleTalk routing module to it's initial state. This command is equivalent to disabling and then enabling the AppleTalk routing module.

Examples To reset the AppleTalk routing module, use the command:

```
RESET APPLE
```

See Also DISABLE APPLE
ENABLE APPLE

SET APPLE PORT

Syntax

```
SET APPLE PORT=port [DEMAND={ON|OFF|YES|NO}]
   [HINT=network:node] [SEED=seed]
```

where:

- *port* is an AppleTalk port number in the range 1 to 127.
- *network* is an AppleTalk network number, in the range 0 to 65279.
- *node* is decimal number in the range 0 to 127.
- *seed* is an AppleTalk network number range of the form of "nnnnnnn-nnnnnnn". A seed specified as "aaa" is treated as "aaa-aaa".

Description

This command modifies the configuration of an AppleTalk port.

The PORT parameter specifies the AppleTalk port to be modified. The AppleTalk port must already exist.

The DEMAND parameter specifies whether or not the port is connected to a dial-on-demand PPP link that will automatically be disconnected when there has been no traffic for a period set by the IDLE parameter of the associated PPP interface. AppleTalk routing broadcasts are suppressed on AppleTalk ports with the DEMAND parameter set to ON. The default is OFF.

The HINT parameter specifies the network and node to be used for the port if the network number falls within the cable range and the node number has not been used by any other device in the network. This parameter is only valid for ETH and PPP interfaces and is required for AppleTalk ports using numbered PPP interfaces. By default, AppleTalk uses addressless PPP interfaces.

The SEED parameter identifies the router as a seed router for the AppleTalk network with the specified network number. The SEED parameter is only valid for Ethernet interfaces.

The SEED and HINT parameters can be used in combination to influence the way AppleTalk assigns AppleTalk node numbers to devices on attached AppleTalk networks (Table 21-3 on page 21-17). If HINT is specified with a value of 0 for the network number (e.g. 0:8), then AppleTalk will attempt to select a network number (w) in the range specified by SEED with a free matching node number (8). If HINT is specified with a non-zero value for both the network and node number (e.g. 2:8) then AppleTalk will attempt to use the specified network and node number (2:8). If HINT is specified with a value of 0 for the node number (e.g. 2:0), then AppleTalk will use the specified network number (provided it is in the range specified by SEED) and select with a free node number (nnn) in that network. If SEED is not specified, HINT will not accept non-zero network numbers. In this case AppleTalk will select a network number (mmm) from the network numbers already configured on the AppleTalk network. In all cases, if the required node number is in use, the next free node number in the same network is used. If all node numbers in the network are used, AppleTalk will try other network numbers in the range specified by SEED.

The RESET APPLE command should be executed after a SET APPLE PORT command to ensure that changes take effect immediately.

Examples To set the seed for port 2 to 123-125, use the command:

```
SET APPLE PORT=2 SEED=123-125
```

See Also ADD APPLE PORT
DELETE APPLE PORT
SHOW APPLE PORT

SET APPLE ROUTECONVERT

Syntax `SET APPLE ROUTECONVERT={ON|OFF|YES|NO}`

Description This command enables or disables the conversion of non-extended (LocalTalk) network routes to extended network routes when RTMP packets are generated. Route conversion is necessary on any internet with a LocalTalk network that must communicate with other AppleTalk networks on the same internet. By default route conversion is disabled.

A *nonextended* network (e.g. a LocalTalk network) is assigned exactly one network number and can have a maximum of 254 active nodes, each uniquely identified by each node's 8-bit AppleTalk node ID. An *extended* network may have multiple network numbers and up to approximately 16 million nodes, each uniquely identified by network number/node ID pairs. An extended network can be thought of as a number of nonextended networks all residing on the same physical data link (e.g. a LAN with multiple network numbers accessed via a WAN link).

Examples To enable the conversion of non-extended network routes to extended network routes, use the command:

```
SET APPLE ROUTECONVERT=ON
```

See Also SHOW APPLE

SET APPLE ZONE

Syntax SET APPLE ZONE=*zone-name* PORT=*port* [DEFAULT]

where:

- *zone-name* is a character string, 1 to 32 characters in length. Valid characters are lowercase letters (a–z), uppercase letters (A–Z) and decimal digits (0–9).
- *port* is an AppleTalk port number in the range 1 to 127.

Description This command sets the default zone for the AppleTalk network to which the port is connected. If the router is a seed router for the network attached to the port, the port must have a default zone, specified by the DEFAULT parameter.

The DEFAULT parameter identifies the specified zone name as the default zone for the network to which the port is connected.

The PORT parameter specifies the AppleTalk port for which the zone is to be the default zone. The AppleTalk port must already exist.

The ZONE parameter specifies the name of the zone to be the default zone for the AppleTalk network attached to the port. Only ports connected to AppleTalk networks (i.e. Ethernet interfaces) support zone names.

Examples To set zone "Godzilla" on port 2 to be the default zone, use the command:

```
SET APPLE ZONE=Godzilla PORT=2 DEFAULT
```

See Also ADD APPLE ZONE
DELETE APPLE ZONE
SHOW APPLE ZONE

SHOW APPLE

Syntax SHOW APPLE

Description This command displays the current status of the AppleTalk routing module (Figure 21-3 on page 21-26, Table 21-4 on page 21-26).

Figure 21-3: Example output from the SHOW APPLE command.

```
Appletalk Module Configuration
----------------------------------------
Module Status .................. Enabled
LocalTalk Route Conversion ..... On
----------------------------------------
```

Table 21-4: Parameters displayed in the output of the SHOW APPLE command.

Parameter	Meaning
Module Status	The status of the AppleTalk routing module; one of "Enabled" or "Disabled".
LocalTalk Route Conversion	Whether or not LocalTalk non-extended routes are converted to extended routes in RTMP packets; one of "On" or "Off".

Examples To show the status of the AppleTalk routing module, use the command:

```
SHOW APPLE
```

See Also ENABLE APPLE
DISABLE APPLE

SHOW APPLE AARP

Syntax SHOW APPLE AARP

Description This command displays the contents of the AARP cache (Figure 21-4 on page 21-27, Table 21-5 on page 21-27).

Figure 21-4: Example output from the SHOW APPLE AARP command.

```
Apple Address Resolution Table

  Network  Node  Physical
  --------------------------------
  22       124   00-00-c0-c9-c6-7b
  --------------------------------
```

Table 21-5: Parameters displayed in the output of the SHOW APPLE AARP command.

Parameter	Meaning
Network	The AppleTalk network number of the AppleTalk device.
Node	The AppleTalk node number of the AppleTalk device.
Physical	The physical (MAC) address of the AppleTalk device on the network.

Examples To display the current AARP cache, use the command:

```
SHOW APPLE AARP
```

SHOW APPLE CIRCUIT

Syntax SHOW APPLE CIRCUIT

Description This command displays all MIOX circuits used and available for use by the AppleTalk routing module (Figure 21-5 on page 21-27, Table 21-6 on page 21-28).

Figure 21-5: Example output from the SHOW APPLE CIRCUIT command.

```
AppleTalk Circuits
--------------------------------------------------
Port  Interface  Circuit Name     Number  State
--------------------------------------------------
   1  x25t0      ToB                   0  Active
```

Table 21-6: Parameters displayed in the output of the SHOW APPLE CIRCUIT command.

Parameter	Meaning
Port	The number of the AppleTalk port.
Interface	The X25T interface used by the AppleTalk port.
Circuit Name	The name of a MIOX circuit defined for the X25T interface.
Number	The PVC or SVC circuit number of the MIOX circuit.
State	The state of the MIOX circuit; one of "Active" or "Allowed". Only active circuits can be used for AppleTalk routing.

Examples To display the available MIOX circuits, use the command:

```
SHOW APPLE CIRCUITS
```

See Also ADD APPLE CIRCUIT
DELETE APPLE CIRCUIT

SHOW APPLE COUNTER

Syntax

```
SHOW APPLE COUNTER={ALL|AARP|ATP|DDP|NBP|PACKET|PKT|PORT |
    ROUTE|RTMP|ZIP|ZONE}
```

Description This command displays all AppleTalk counters or all counters from the specified group of counters.

If AARP is specified, counters for the AppleTalk Address Resolution Protocol (AARP) are displayed (Figure 21-6 on page 21-28, Table 21-7 on page 21-29). If ATP is specified, counters for the AppleTalk Transaction Protocol (ATP) are displayed (Figure 21-7 on page 21-29, Table 21-8 on page 21-29). If DDP, PKT or PACKET is specified, counters for the AppleTalk Datagram Delivery Protocol (DDP) are displayed (Figure 21-8 on page 21-30, Table 21-9 on page 21-30). If NBP is specified, counters for the AppleTalk Name Binding Protocol (NBP) are displayed (Figure 21-9 on page 21-31, Table 21-10 on page 21-31). If PORT is specified, counters for traffic transmitted and received via the AppleTalk ports on the router are displayed (Figure 21-10 on page 21-31, Table 21-11 on page 21-32). If ROUTE or RTMP is specified, counters for the AppleTalk Routing Table Maintenance Protocol (RTMP) are displayed (Figure 21-11 on page 21-32, Table 21-12 on page 21-32). If ZIP or ZONE is specified, counters for the AppleTalk Zone Information Protocol (ZIP) are displayed (Figure 21-12 on page 21-33, Table 21-13 on page 21-33).

Figure 21-6: Example output from the SHOW APPLE COUNTER=AARP command.

```
AARP Counters
-------------
  aarpLookups ...................... 241
  aarpHits ......................... 137
```

Table 21-7: Parameters displayed in the output of the SHOW APPLE COUNT=AARP command.

Parameter	Meaning
aarpLookups	The number of times the AARP cache for this entity was searched.
aarpHits	The number of times an entry was searched for and found in the AARP cache for this entity.

Figure 21-7: Example output from the SHOW APPLE COUNTER=ATP command.

```
ATP Counters
------------

  atpInPkts ......................... 1
  atpOutPkts ........................ 1
  atpTRequestRetransmissions ......... 0
  atpTResponseRetransmissions ........ 0
  atpReleaseTimerExpiredCounts ....... 0
  atpRetryCountExceededs ............. 0
```

Table 21-8: Parameters displayed in the output of the SHOW APPLE COUNT=ATP command.

Parameter	Meaning
atpInPkts	The number of ATP packets received by this entity.
atpOutPkts	The number of ATP packets sent by this entity.
atpTRequestRetransmissions	The number of times that a timeout occurred and a Transaction Request packet needed to be retransmitted by this host.
atpTResponseRetransmissions	The number of times a timeout was detected and a Transaction Response packet needed to be retransmitted by this host.
atpReleaseTimerExpiredCounts	The number of times the release timer expired, as a result of which a Request Control Block had to be deleted.
atpRetryCountExceededs	The number of times the retry count was exceeded, and an error was returned to the client of ATP.

Figure 21-8: Example output from the SHOW APPLE COUNTER=DDP command.

```
DDP Counters
------------
  ddpOutRequests ................. 6285
  ddpOutShorts ...................... 0
  ddpOutLongs .................... 6303
  ddpInReceives .................. 6250
  ddpInLocalDatagrams ............ 6231
  ddpNoProtocolHandlers ............. 4
  ddpTooShortErrors ................. 0
  ddpTooLongErrors .................. 0
  ddpShortDDPErrors ................. 0
  ddpChecksumErrors ................. 0
  ddpForwRequests .................. 18
  ddpOutNoRoutes .................... 1
  ddpBroadcastErrors ................ 0
  ddpHopCountErrors ................. 0
```

Table 21-9: Parameters displayed in the output of the SHOW APPLE COUNT=DDP command.

Parameter	Meaning
ddpOutRequests	The total number of DDP datagrams which were supplied to DDP by local clients in request for transmission. Note that this counter does not include any datagrams counted in ddpForwRequests.
ddpOutShorts	The total number of short DDP datagrams which were transmitted from this entity.
ddpOutLongs	The total number of long DDP datagrams which were transmitted from this entity.
ddpInReceives	The total number of input datagrams received by DDP, including those received in error.
ddpInLocalDatagrams	The total number of input DDP datagrams for which this entity was their final DDP destination.
ddpNoProtocolHandlers	The total number of DDP datagrams addressed to this entity that were addressed to an upper layer protocol for which no protocol handler existed.
ddpTooShortErrors	The total number of input DDP datagrams dropped because the received data length was less than the data length specified in the DDP header or the received data length was less than the expected DDP header.
ddpTooLongErrors	The total number of input DDP datagrams dropped because they exceeded the maximum DDP datagram size.
ddpShortDDPErrors	The total number of input DDP datagrams dropped because this entity was not their final destination and their type was short DDP.
ddpChecksumErrors	The total number of input DDP datagrams for which this DDP entity was their final destination, and which were dropped because of a checksum error.
ddpForwRequests	The number of input datagrams for which this entity was not their final DDP destination, as a result of which an attempt was made to find a route to forward them to that final destination.

Table 21-9: Parameters displayed in the output of the SHOW APPLE COUNT=DDP command. (Continued)

Parameter	Meaning
ddpOutNoRoutes	The total number of DDP datagrams dropped because a route could not be found to their final destination.
ddpBroadcastErrors	The total number of input DDP datagrams dropped because this entity was not their final destination and they were addressed to the link level broadcast.
ddpHopCountErrors	The total number of input DDP datagrams dropped because this entity was not their final destination and their hop count would exceed 15.

Figure 21-9: Example output from the SHOW APPLE COUNTER=NBP command.

```
NBP Counters
------------

  nbpInLookUpRequests ................ 0
  nbpInLookUpReplies ................. 0
  nbpInBroadcastRequests ............ 65
  nbpInForwardRequests ............... 0
  nbpOutLookUpReplies ................ 0
  nbpRegistrationFailures ............ 0
  nbpInErrors ........................ 0
```

Table 21-10: Parameters displayed in the output of the SHOW APPLE COUNT=NBP command.

Parameter	Meaning
nbpInLookUpRequests	The number of NBP LookUp Requests received.
nbpInLookUpReplies	The number of NBP LookUp Replies received.
nbpInBroadcastRequests	The number of NBP Broadcast Requests received.
nbpInForwardRequests	The number of NBP Forward Requests received.
nbpOutLookUpReplies	The number of NBP LookUp Replies sent.
nbpRegistrationFailures	The number of times this node experienced a failure in attempting to register an NBP entity.
nbpInErrors	The number of NBP packets received by this entity that were rejected for any error.

Figure 21-10: Example output from the SHOW APPLE COUNTER=PORT command.

```
Port Counters
-------------

  Port        atportInPkts        atportOutPkts
  ---------------------------------------------
     1                6247                 6402
     2                   7                    8
  ---------------------------------------------
```

Table 21-11: Parameters displayed in the output of the SHOW APPLE COUNT=PORT command.

Parameter	Meaning
atportInPkts	The number of packets received by this entity on this port.
atportOutPkts	The number of packets transmitted by this entity on this port.

Figure 21-11: Example output from the SHOW APPLE COUNTER=ROUTE command.

```
RTMP Counters
-------------

  rtmpInDataPkts .................. 6163
  rtmpOutDataPkts ................. 6092
  rtmpInRequestPkts .................. 0
  rtmpNextIREqualChanges ......... 18481
  rtmpNextIRLessChanges .............. 1
  rtmpRouteDeletes ................... 0
  rtmpRoutingTableOverflows .......... 0
  rtmpOutRequestPkts ................. 0
  rtmpInVersionMismatches ............ 0
  rtmpInErrors ....................... 0
```

Table 21-12: Parameters displayed in the output of the SHOW APPLE COUNT=ROUTE command.

Parameter	Meaning
rtmpInDataPkts	The number of good RTMP data packets received by this entity.
rtmpOutDataPkts	The number of RTMP packets sent by this entity.
rtmpInRequestPkts	The number of good RTMP Request packets received by this entity.
rtmpNextIREqualChanges	The number of times RTMP changed the Next Internet Router in a routing entry because the hop count advertised in a routing tuple was equal to the current hop count for a particular network.
rtmpNextIRLessChanges	The number of times RTMP changed the Next Internet Router in a routing entry because the hop count advertised in a routing tuple was less than the current hop count for a particular network.
rtmpRouteDeletes	The number of times RTMP deleted a route because it was aged out of the table. This can help to detect routing problems.
rtmpRoutingTableOverflows	The number of times RTMP attempted to add a route to the RTMP table but failed due to lack of space.
rtmpOutRequestPkts	The number of RTMP Request packets sent by this entity.
rtmpInVersionMismatches	The number of RTMP packets received by this entity that were rejected due to a version mismatch.
rtmpInErrors	The number of RTMP packets received by this entity that were rejected for an error other than version mismatch.

Figure 21-12: Example output from the SHOW APPLE COUNTER=ZIP command.

```
RTMP Counters
-------------

  rtmpInDataPkts .................. 6163
  rtmpOutDataPkts ................. 6092
  rtmpInRequestPkts .................. 0
  rtmpNextIREqualChanges ......... 18481
  rtmpNextIRLessChanges .............. 1
  rtmpRouteDeletes ................... 0
  rtmpRoutingTableOverflows .......... 0
  rtmpOutRequestPkts ................. 0
  rtmpInVersionMismatches ............ 0
  rtmpInErrors ....................... 0
```

Table 21-13: Parameters displayed in the output of the SHOW APPLE COUNT=ZIP command.

Parameter	Meaning
zipInZipQueries	The number of ZIP Queries received by this entity.
zipInZipReplies	The number of ZIP Replies received by this entity.
zipInZipExtendedReplies	The number of ZIP Extended Replies received by this entity.
zipZoneConflictErrors	The number of times a conflict was been detected between this entity's zone information and another entity's zone information.
zipInObsoletes	The number of ZIP Takedown or ZIP Bringup packets received by this entity. Note that as the ZIP Takedown and ZIP Bringup packets have been obsoleted, the receipt of one of these packets indicates that a node sent it in error.
zipInGetNetInfos	The number of ZIP GetNetInfo packets received on this port by this entity.
zipOutGetNetInfoReplies	The number of ZIP GetNetInfo Reply packets sent out this port by this entity.
zipZoneOutInvalids	The number of times this entity has sent a ZIP GetNetInfo Reply with the zone invalid bit set in response to a GetNetInfo Request with an invalid zone name.
zipAddressInvalids	The number of times this entity had to broadcast a ZIP GetNetInfo Reply because the GetNetInfo Request had an invalid address.
zipOutGetNetInfos	The number of ZIP GetNetInfo packets sent out this port by this entity.
zipInGetNetInfoReplies	The number of ZIP GetNetInfo Reply packets received on this port by this entity.
zipZoneInInvalids	The number of times this entity has received a ZIP GetNetInfo Reply with the zone invalid bit set because the corresponding GetNetInfo Request had an invalid zone name.
zipInErrors	The number of ZIP packets received by this entity that were rejected for any error.

Examples

To display the PORT counter group, use the command:

```
SHOW APPLE COUNT=PORT
```

See Also

SHOW APPLE
SHOW APPLE AARP
SHOW APPLE CIRCUIT
SHOW APPLE DLCI
SHOW APPLE PORT
SHOW APPLE ROUTE
SHOW APPLE ZONE

SHOW APPLE DLCI

Syntax

```
SHOW APPLE DLCI
```

Description

This command displays all the DLCIs used and available for use by the AppleTalk routing module (Figure 21-13 on page 21-34, Table 21-14 on page 21-34).

Figure 21-13: Example output from the SHOW APPLE DLCI command.

```
Port   Interface   DLCI    Allowed   Active   Usable
----------------------------------------------------
  3    fr0           23    Yes       No       No
                     70    Yes       No       No
----------------------------------------------------
```

Table 21-14: Parameters displayed in the output of the SHOW APPLE DLCI command.

Parameter	Meaning
Port	The number of the AppleTalk port.
Interface	The Frame Relay interface used by the AppleTalk port.
DLCI	The DLCI of a DLC defined for the Frame Relay interface.
Allowed	Whether or not the circuit is allowed to be used by the AppleTalk routing module; one of "Yes" or "No".
Active	Whether or not Frame Relay indicates the circuit is active; one of "Yes" or "No".
Usable	Whether or not the circuit can be used by the AppleTalk routing module; one of "Yes" (the circuit is allowed and active) or "No".

Examples

To display all available DLCIs, use the command:

```
SHOW APPLE DLCI
```

See Also

ADD APPLE DLCI
DELETE APPLE DLCI

SHOW APPLE PORT

Syntax SHOW APPLE PORT[=*port*]

where:

- *port* is an AppleTalk port number in the range 1 to 127.

Description This command displays information about AppleTalk ports. If a port number is specified, detailed information about the specified port is displayed (Figure 21-14 on page 21-35, Table 21-15 on page 21-35). If a port number is not specified, summary information about all ports is displayed (Figure 21-15 on page 21-35, Table 21-15 on page 21-35).

Figure 21-14: Example output from the SHOW APPLE PORT command for a specified port.

```
Appletalk Port Details
------------------------------------
  Port Number ............. 1
  Interface ................ eth0
  ifIndex .................. 1
  Node ID .................. 217
  Network Number ........... 22
  Network Range Start ...... 22
  Network Range End ........ 22
  State .................... ACTIVE
  Seed ..................... NO
  Seed Network Start ....... 0
  Seed Network End ......... 0
  Hint ..................... YES
  Hint Node ID ............. 179
  Hint Network ............. 22
  Default Zone ............. -

  Zone List is Empty
------------------------------------
```

Figure 21-15: Example output from the SHOW APPLE PORT command.

```
Appletalk Port Summary

  Port  Interface  Node ID  Network Address  State       Demand  Link  Route
  ---------------------------------------------------------------------------
  1     eth0       5        700              ACTIVE      -       -     -
  2     ppp0       0        0                ACTIVE      On      Down  Init
  ---------------------------------------------------------------------------
```

Table 21-15: Parameters displayed in the output of the SHOW APPLE PORT command.

Parameter	Meaning
Port Number	The number of the AppleTalk port.
Interface	The interface used by the port.
ifIndex	The value of ifIndex for the AppleTalk port.
Node ID	The node ID of the port.

Table 21-15: Parameters displayed in the output of the SHOW APPLE PORT command. (Continued)

Parameter	Meaning
Network Number	The network number for the port.
Network Range Start	The first network number in the range of network numbers assigned to the network to which the port is attached.
Network Range End	The last network number in the range of network numbers assigned to the network to which the port is attached.
Network Address	The network address of the port.
State	the state of the port; one of "INITIAL", "PROVISIONAL", "ACTUAL" or "ACTIVE".
Demand	Whether or not the port is configured for dial-on-demand routing; one of "On" or "Off". This parameter is only valid for PPP interfaces.
Link	The status of the link layer interface; one of "Up" (link layer interface is up), "RTMP" (RTMP packets have been transmitted via this port) or "Down" (link layer interface is down). This parameter is only valid for PPP interfaces.
Route	The status of peer route updates; one of "Init" (no route updates have been seen), "RTMP" (RTMP packets have been seen since the link came up) or "Updt" (route updates have been seen since the first RTMP packet). This parameter is only valid for PPP interfaces.
Seed	Whether or not this router is a seed router for the network to which the port is attached; one of "YES" or "No".
Seed Network Start	If the *Seed* field is "YES", the first network number in the range of seed network numbers for the network to which the port is attached.
Seed Network End	If the *Seed* field is "YES", the last network number in the range of seed network numbers for the network to which the port is attached.
Hint	Whether or not this router has a hint protocol address that can be used in the address acquisition phase at startup; one of "YES" or "NO".
Hint Node ID	If the *Hint* field is "YES", the node ID portion of the hint protocol address for this router.
Hint Network	If the *Hint* field is "YES", the network number portion of the hint protocol address for this router.
Default Zone	The default zone, if any, for the network to which this port is attached.
Zone List	The lists of zones, if any, defined for the network to which this port is attached.

Examples To display all AppleTalk ports, use the command:

```
SHOW APPLE PORT
```

See Also ADD APPLE PORT
DELETE APPLE PORT

SHOW APPLE ROUTE

Syntax `SHOW APPLE ROUTE`

Description This command displays the contents of the AppleTalk routing table, including both dynamic and static routes (Figure 21-16 on page 21-37, Table 21-16 on page 21-37).

Figure 21-16: Example output from the SHOW APPLE ROUTE command.

```
AppleTalk routing table

  Network range   Hops   Port   Interface   Next hop   Type
  -----------------------------------------------------------
  21              1       1     eth0        22:124     Remote
  22              0       1     eth0        0:0        Direct
  23              1       1     eth0        22:124     Remote
  200             2       2     ppp0        -          Static
  50000           1       1     eth0        22:124     Remote
  -----------------------------------------------------------
```

Table 21-16: Parameters displayed in the output of the SHOW APPLE ROUTE command.

Parameter	Meaning
Network range	The network or network range of the destination network.
Hops	The number of hops to the destination network.
Port	The AppleTalk port via which the destination network can be reached.
Interface	The lower layer interface used by the AppleTalk port.
Next hop	The AppleTalk network number and node ID of the next router on the route to the destination network.
Type	The type of route; one of "Remote" (dynamically learned route to a remote network), "Direct" (dynamically learned route to a network to which the router is directly connected) or "Static".

Examples To display the current contents of the AppleTalk routing table, use the command:

```
SHOW APPLE ROUTE
```

See Also ADD APPLE ROUTE
DELETE APPLE ROUTE

SHOW APPLE ZONE

Syntax `SHOW APPLE ZONE`

Description This command displays information about all currently known and configured zones (Figure 21-17 on page 21-38, Table 21-17 on page 21-38).

Figure 21-17: Example output from the SHOW APPLE ZONE command.

```
Zone Information Table

  Zone                              Port  Network
  ------------------------------------------------
  Aslan Internal                    -     50000-50000
  BrickBat                          -     100-100
  Finance                           -     21-21
  Finance                           2     22-22
  ------------------------------------------------
```

Table 21-17: Parameters displayed in the output of the SHOW APPLE ZONE command.

Parameter	Meaning
Zone	The name of the AppleTalk Zone.
Port	The AppleTalk port with which the zone is associated.
Network	The network number range(s) associated with the zone.

Examples To display all AppleTalk zones, use the command:

```
SHOW APPLE ZONE
```

See Also ADD APPLE ZONE
DELETE APPLE ZONE