CHAPTER 9

IPv6 over ATM

ATM networks[1], for their connection-oriented nature, don't provide an ideal environment for connectionless network protocols such as IPv4, IPv6, IPX, Decnet, and so on. A possible solution for a layer 3 protocol to be supported by an ATM network cannot even be foreseen with acceptable performance. On the one hand, it is true that in the near future, many intranets will probably continue to be multi-protocol and therefore need to transmit and to receive, besides IP packets, other protocols (such as Decnet, IPX, OSI); on the other hand, it is equally true that the only protocol that is worth modifying further to suit ATM is IP (both version 4 and version 6) for the major role it will have in the future of networks. Originally, a classification of IP over ATM approaches was tried, by differentiating them based on their geographic extension (LAN, MAN, and WAN).

This classification was discontinued as improper; in ATM networks, the distance increases the propagation delay and reduces performance, but it doesn't substantially change the network organization and packet routing problems.

The use of an ATM network to transport IPv6 packets can be relatively simple or very complex, depending on how the ATM network itself is used. Many commercial proposals for ATM WANs (wide area networks) offer a service based on *PVCs* (Permanent Virtual Connections) and an internetworking between local networks and the wide area network implemented through routers. This method of using ATM doesn't present particular problems because routers see PVCs as point-to-point channels. This approach is frequently chosen when

- Internetworking sizes are significant
- Heterogeneous transmission media are used, making the use of a unique network technology impossible
- Reliability reasons impose a partially meshed technology, also with heterogeneous transmission media

The only decision to make is how to segment IP packets into ATM cells, but standard solutions are already available for this problem.

The situation is different if we want to use *SVCs* (Switched Virtual Connections), which are activated through UNI (*User to Network Interface*)[2] signaling procedures. SVCs make ATM a multi-access network—that is, a network in which all other users of the network can be reached from any connection point.

Also, LANs are multi-access networks, which are different from ATM for their connectionless nature and because they offer a native support to the broadcast traffic. The lack of a mechanism to transmit the broadcast traffic classifies ATM as an *NBMA (Non Broadcast Multiple Access)* network technology. Other NBMA network technologies have been available for many years—for example, those based on X.25 and Frame Relay protocols—but the transport of IP packets on NBMA networks acquires a particular relevance only with ATM. In fact, market analysis agrees that, in the near future, both ATM and IPv6 will be widespread technologies, and therefore we must find efficient ways to use them jointly.

The use of SVC requires mechanisms in which the IPv6 protocol activates UNI signaling procedures to create and terminate SVCs, mechanisms that are in contrast with the connectionless nature of the IP protocol.

Moreover, the lack of a native support for the broadcast is particularly important for the Neighbor Discovery protocol (see Chapter 6), which is based on the assumption that the link level underlying IPv6 can support multicast transmissions.

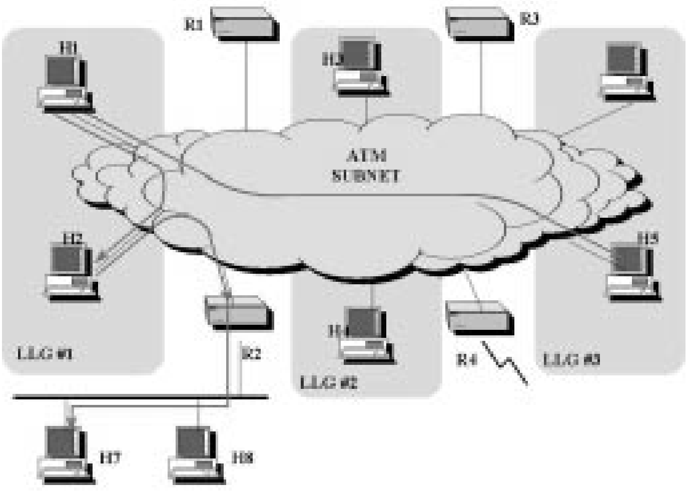
Looking to the future of networks and of internetworking, we will see an ever-growing number of ATM networks interconnected at the ATM level—that is, through connections between switches. This structure creates the possibility of setting SVCs between whichever couples of nodes can pass IP subnet limits; however, doing so violates the classic IP model in which distinct IP subnets can communicate between them only through routers.

Problems relevant to IP over ATM internetworking can be better understood by analyzing Figure 9-1, in which IP subnets are identified by the acronym *LLG* (*Logical Link Group*), according with the terminology proposed for IPv6 on ATM.

From the analysis of Figure 9-1, we can understand how much the problem of routing IP over ATM is complicated by the possibility of setting SVCs between two stations directly connected to ATM even if belonging to different LLGs (for example, H1 and H5), implementing a process called *cut-through routing*. Another problem that needs an efficient solution is the identification of the best exit router (*egress router*) toward a station not connected to ATM (for example, the router R2 for the communication between H2 and H7).

Of course, having cut-through routing schemes to use IPv6 on ATM is not necessary; we can still use the classical IP routing approach and cross routers following IP routing rules (in Figure 9-1, for going from H1 to H5, the classical IP routing can occur along the path H1 - R1 - R3 - H5). Cut-

Figure 9-1
IPv6 over ATM



through routing becomes necessary with the growth of network sizes because the number of routers to be traversed can become high, penalizing the performance greatly.

In the following text, we will see how the solution to some problems is already consolidated, based on solutions standardized for IPv4 on ATM; whereas the solution to other problems is currently the subject of further discussion. For this reason, the remaining part of the chapter is subdivided into Section 9.1, which describes the more consolidated aspects, and into Section 9.2, which describes those not yet completely defined. In Section 9.3, we will discuss alternative approaches that don't use UNI and P-NNI signaling procedures.

9.1 Defined Aspects

Defined aspects deal with packet encapsulation, the identification of VC (Virtual Connection) endpoints, and modalities to transport IPv6 packets in ATM cells.

Solutions to these problems are common to all proposals of IPv6 on ATM and are independent of topology or routing considerations and of the use of PVCs or SVCs.

An example of interconnection of two hosts and an IPv6 router through an ATM network (ATM subnet) is shown in Figure 9-2.

The problem of the encapsulation and of the identification of VC endpoints is treated by RFC 1483[3], which provides a multi-protocol solution, valid also for IPv6. RFC 1483 provides two possible solutions: LLC/SNAP encapsulation and VC multiplexing.

The problem of transporting IPv6 packets in ATM cells is solved by adopting the AAL5 (ATM Adaptation Layer 5).

9.1.1 LLC/SNAP Encapsulation

RFC 1483[3] proposes *LLC/SNAP* encapsulation as the default solution. This approach is an adaptation to ATM of the solution developed in project IEEE 802[4]. It allows the transportation of an arbitrary number of protocols within a single VC, identifying them by means of an LLC/SNAP header (see Figure 9-3).

Figure 9-4 shows an example of several Ethernet-derived protocols (OUI = 00-00-00H) that share the same VC and that are differentiated by the value of the PID (*Protocol IDentifier*) field.

The LLC/SNAP encapsulation is used both for IPv6 unicast packets, for multicast packets, and also for the interaction between IPv6 stations and the MARS (*Multicast Address Resolution Server*)[5], described in Section 9.2.4.

In the case of IPv6 unicast packets, the encapsulation used is exactly the one shown in Figure 9-3. In contrast, IPv6 packets sent to the MARS are enveloped by using the OUI 0x00-00-5E registered by the IANA. In the case of control messages, the PID 0x00-03 is used, as shown in Figure 9-5.

A more complex description is needed for multicast IPv6 packets (possibly relayed through an MCS, see Section 9.2.4) that must be encapsulated as shown in Figure 9-6. The presence of the field pkt$cmi (*CMI: Cluster Member ID*) within these packets allows a station to recognize, among received multicast messages, those it transmitted; therefore, it will not to process them. The field pkt$pro (packet protocol) indicates the protocol that generated the encapsulated PDU (IPv6 in the case of Figure 9-6).

Figure 9-2
Interconnection of IP hosts through ATM

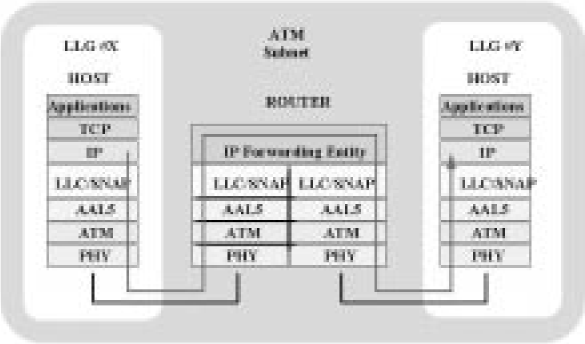
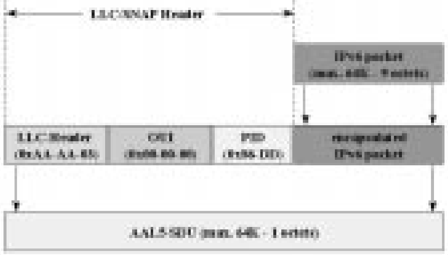


Figure 9-3
LLC/SNAP encapsulation



9.1.2 VC Multiplexing

The UNI^2 standard provides that the endpoint of a VC is set during the call setup phase. A simple approach is to use the *VC multiplexing* or *null encapsulation* that provides for termination of a VC through an AAL5 instance directly on a layer 3 protocol (see Figure 9-7). When the VC multiplexing is used in IPv6, the end of the VC is the IPv6 protocol itself; that is, the IPv6 packet is directly placed inside the AAL5-SDU.

This approach is restrictive in multi-protocol environments in which each protocol requires the creation of a separate VC; it causes a considerable load on ATM switches for the signaling associated with the opening and closing of VCs. Moreover, the number of VCs is very high, and it can exceed the maximum number of VCs admitted by switches.

Figure 9-4
Sharing a VC through LLC/SNAP

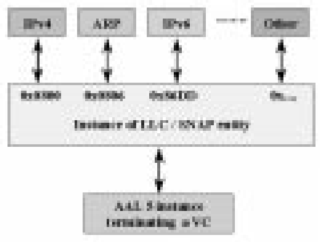

IPv4
ARP
IPv6
Other
0x0800
0x0806
0x86DD
0x...
Instance of LLC / SNAP entity
AAL 5 instance terminating a VC

Figure 9-5
Encapsulation of a MARS control message

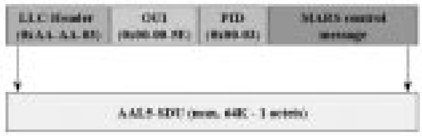
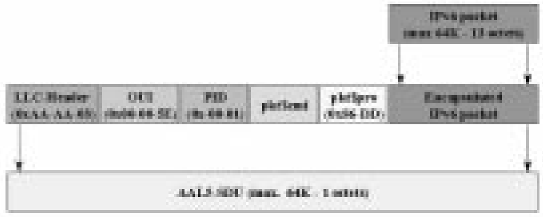


Figure 9-6
LLC/SNAP encapsulation for multicast packets



9.1.3 AAL Type 5

Both the preceding solutions assume that the packet is segmented using AAL5 (see **1** and **3**). This AAL has been standardized by the ATM Forum, starting from a proposal to simplify AAL3/4, called SEAL (Simple and Efficient Adaptation Layer). AAL5 is designed to offer only a connectionless service. Today AAL5 has been adopted worldwide to make data transmission very simple and efficient. The simplification is drastic, both for what relates to the CS sublayer (Convergence Sublayer), which has been emptied in practice, and for what relates to the SAR (Segmentation And Reassembly) sublayer.

In preceding sections, we saw how an IPv6 packet is enveloped in an AAL5-SDU. The AAL5 adds a PAD field to the AAL5-SDU to normalize the length of the AAL5-PDU to a multiple of 48 octets, a control field also containing the length of the AAL5-PDU, and a CRC on 32 bits computed on the PDU itself.

The AAL5-PDU is subdivided into a sequence of 48-octet segments (SAR-PDU) that are neither numbered nor identified in any way (see Figure 9-8).

The SAR-PDU, shown in Figure 9-9, is 48 octets long and coincides with the payload of the ATM cell. The last segment is marked by the setting of a bit in the PT (*Payload Type*) field of the header of the ATM cell transporting it.

When a cell, whose bit is set in the PT, is received by the SAR sublayer of the AAL 5, the SAR sublayer assembles all the received SAR-PDUs rebuilding the AAL5-PDU, and it verifies the length and the CRC (refer to Figure 9-8). If the AAL5-PDU is valid, the AAL5-SDU is extracted from it; and from this, the IPv6 packet. In case of errors, the AAL5-PDU is discarded without any other action, like happens at the MAC level in the case of an erroneous Ethernet frame.

Figure 9-7
Multiprotocol networks through VC multiplexing

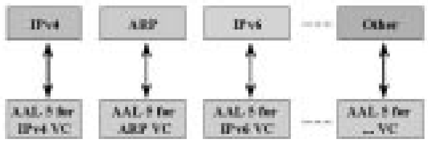


Figure 9-8
Process of AAL5 segmentation and reassembling

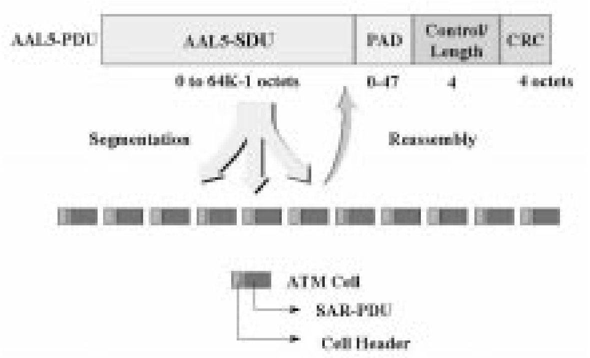
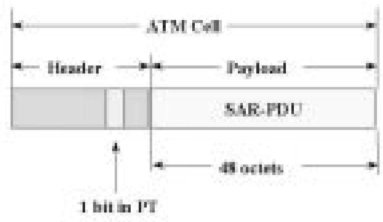


Figure 9-9
Format of the AAL5 SAR-PDU



9.2 Work in Progress

Most of the techniques described in the following subsections will certainly be part of the solution or solutions that will be standardized for IPv6 on ATM. Some of these techniques are already included in some RFCs; others have been widely discussed by IETF working groups. Currently, what is not already clear is how different techniques will combine to provide the standard solution or solutions.

9.2.1 Neighbor Discovery

The Neighbor Discovery (ND) protocol, described in Chapter 6, is not easily adaptable to ATM networks because it assumes that the underlying link level supports multicast transmissions and differentiates on-link and off-link stations, and also because it doesn't explicitly deal with cut-through routing problems[6].

The need for cut-through routing derives from the inadequacy of the concepts of on-link and off-link when large ATM networks are deployed. The concept of link is replaced by the concept of LLG (*Logical Link*

Group), a set of stations that share the same IPv6 address prefix and that are therefore neighbors. Many LLGs can or must be configured on the same ATM network for technical and administrative reasons. Given two IPv6 nodes, we can have the following three cases:

- *On LLG Neighbor:* Two nodes connected to the same ATM network and belonging to the same LLG. This case is the simplest one because it follows the normal way of operating for IPv6. An example is the connection between hosts H1 and H2 in Figure 9-1.
- *Off LLG Neighbor:* Two nodes connected to the same ATM network but not belonging to the same LLG. When two nodes are Off LLG Neighbor, the cut-through routing can be performed between them. An example of this situation is the connection between hosts H1 and H5 in Figure 9-1.
- *Off LLG not Neighbor:* Two nodes that are not connected to the same ATM network and that therefore cannot belong to the same LLG. When two nodes are Off LLG not Neighbor, a direct VC cannot be activated between them, but the best egress router can be determined and a cut-through toward it can be activated. An example of this situation is a connection between hosts H2 and H7 in Figure 9-1.

A simplified solution to ND problems is to use a MARS service (see Section 9.2.4) to emulate generalized multicast support and therefore allow the ND to operate like on a LAN. Note that this solution is a further use of MARS; in fact, MARS has mainly been developed to manage layer 3 multicast addresses (see Section 4.8) like those used by multimedia applications.

The use of MARS solves the problem only for the On LLG Neighbor case, but it doesn't allow cut-through routing. To overcome this limit, a more advanced version[7] has been proposed to provide the creation of an ND server's hierarchy (basically MARS servers devoted to ND problems) in which each server can provide direct answers to the On LLG Neighbor case, while exploiting the hierarchical interconnection with other servers for Off LLG cases.

An alternative proposal[8] is to solve ND problems by reusing the huge amount of work already done to allow the cut-through routing in IPv4, using the NHRP protocol (see Section 9.2.5). This proposal also poses a solution to the problem of the autoconfiguration of IPv6 addresses associated with ATM interfaces (see Section 9.2.2).

A third proposal[9] suggests the use of MARS/MCS within the LLG and

NHRP for the cut-through routing. This proposal introduces the concept of *Transient Neighbors*—that is, temporary neighbors created through ICMP Redirect messages (see Section 9.2.5).

9.2.2 Address Autoconfiguration

The autoconfiguration problem of IPv6 addresses associated with ATM interfaces is complicated by the lack of a multicast native mechanism that allows use of the Duplicate Address Detection procedure (see Section 6.7.4), but also by the presence of the concept of logic interface in ATM. In fact, on an ATM network board, many ATM logical interfaces can be configured, obviously having different addresses (*interface tokens*, according to the IPv6 terminology). The Link Local address autoconfiguration therefore becomes more complex than in the case of LANs where 48-bit MAC addresses are used as interface tokens. This issue raises both the problem of using a number of bits sufficient to univocally identify the interface to avoid duplicated addresses and the problem of using a number of bits sufficient for the network prefix.

This problem does not have a general solution so far. A proposal limited to the NHRP case is described in the IETF Internet Draft *IPv6 over NBMA Networks*[8].

9.2.3 ICMP Redirect

The ICMP Redirect message, which is provided by RFC 1885[10], must be correctly supported by all IPv6 nodes (see Section 5.5.8). Its semantic is extended if compared to the IPv4 one because it allows creation of *Transient Neighbors*—that is, nodes that are temporarily considered neighbors. This capability can be useful in the Off LLG Neighbor case because the ICMP Redirect message can transport the Link Source/Target Address option (see Section 5.5.10). This option can be used to carry the ATM address (on 20 octets) of the target node and therefore to allow the source node to open a dedicated VC with the target node through UNI signaling, by implementing the cut-through routing.

9.2.4 MARS (Multicast Address Resolution Server)

In the introduction, we pointed out the lack of native support for broadcast traffic in ATM because ATM is an NBMA network. The IETF working group “IP over NBMA networks” (formerly “IP over ATM”) released RFC 2022[5] suggesting that the support for the multicast traffic be built by using point-to-multipoint VCs and a MARS (*Multicast Address Resolution Server*).

The MARS is an extension of the ATMARP server standardized for IPv4 in RFC 1577[11]. It implements a recording entity in which layer 3 multicast addresses are associated with ATM interfaces belonging to the multicast group. MARS messages allow the distribution of information about the composition of multicast groups as well as the addition or the cancellation of a node to or from a multicast group. A MARS server administers a point-to-multipoint VC with all nodes that want to receive a multicast support.

A MARS server only keeps track of the composition of multicast groups; it doesn’t attend to the distribution of data packets. Distribution can be made either through an MCS (MultiCast Server) or through a set of point-to-multipoint VCs. In fact, if multicast group A is served by an MCS, the MARS provides the ATM address of the MCS to all the stations that request the resolution of the IPv6 address identifying multicast group A (in Figure 9-10, the address FF15::77). The MCS opens a point-to-multipoint VC with all the stations belonging to the group, and it uses this VC to redistribute multicast data packets.

If the multicast group is not associated with an MCS, the MARS server provides all stations that try to solve the IPv6 multicast address with the list of all ATM addresses associated with the group, and the station creates a dedicated point-to-multipoint VC (see Figure 9-11).

9.2.5 NHRP (Next Hop Resolution Protocol)

A large ATM network is typically subdivided into several independent IP subnets called *LISs* (Logical IP Subnets) in IPv4 and *LLGs* (Logical Link Groups) in IPv6. In IPv4, the ATMARP protocol allows the resolution of the IP address of a destination (host or router) into the corresponding ATM address only if this address belongs to the source LIS. To overcome this limit, the IETF working group called ROLC (Routing Over Large Clouds, which lately joined the group “IP over NBMA networks”) devel-

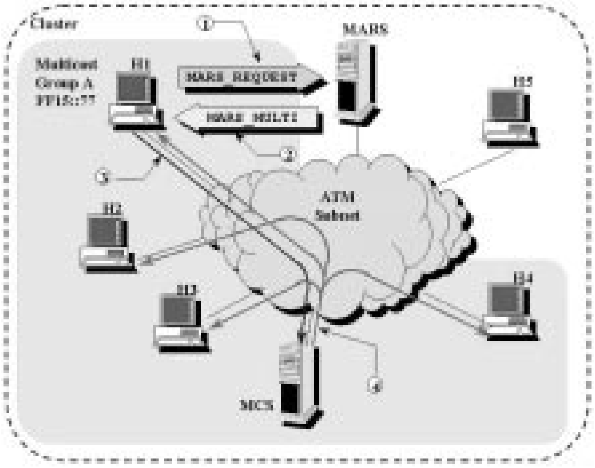


Figure 9-10 MultiCast Server associated with a multicast group

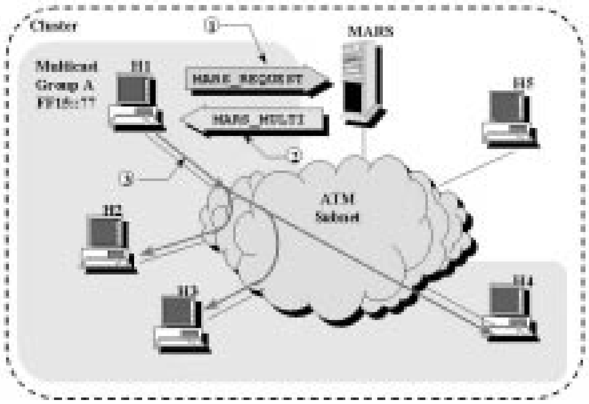


Figure 9-11 A multicast group without MultiCast Server

oped the *NBMA Next Hop Resolution Protocol* (NHRP)[12], a routing and address resolution protocol suitable for all NBMA networking technologies that, like ATM, do not support broadcast transmissions.

NHRP allows a source station (host or router), wanting to communicate over an ATM network, to determine IP and ATM addresses of the *next hop* toward the destination station, given the IP address of the destination station. If the destination is part of the source ATM network, the next hop address returned by NHRP will be the ATM address of the destination itself; otherwise, it will be the address of the router located on the shortest possible path (in terms of layer 3 hops) between source and destination. After the next hop ATM address is known, the source station can open an SVC with it and start the transmission of IP packets. For example, with reference to Figure 9-1, by means of NHRP, H1 can learn the

ATM address of H5 and therefore open an SVC with it instead of sending packets along the multi-hop path H1 - R1 - R3 - H5. Moreover, H2 is informed that the "best" egress router to reach H7 is R2, not the default router R1.

The NHRP protocol, by eliminating from end-to-end paths all unnecessary hops, optimizes remarkably the forwarding process of IP packets within an ATM network.

The NHRP protocol requires the installation, within an ATM network, of one or more entities called *Next Hop Servers* (NHSs). Each NHS serves a determined set of hosts and routers (*clients*). NHSs, besides collaborating among themselves for the resolution of a next hop within their ATM networks, can participate with routing protocols to learn the topology of interconnections.

Each NHS administers a relationship table between IP addresses and ATM addresses of the clients it serves. This table, called the *next hop resolution cache,* can be manually configured or built and dynamically updated in the following ways:

- Through a recording process carried out by clients by sending to their own NHS an NHRP_Register message
- By extracting the information from resolution requests received from clients through the NHRP_Request message
- By extracting the information from replies coming from other network NHSs through the NHRP_Reply message

Let's suppose that station S should determine the ATM address of the next hop toward station D. S addresses its own NHS by sending an NHRP_Request message. The NHRP_Request message is encapsulated in an IP packet and transmitted to the NHS through a VC created at the time of the registration or specifically created for transmitting the request.

In the meanwhile, waiting for the reply from the NHS, S can proceed as follows:

- To drop the packet to be transmitted to D
- To retain the packet until the reply from the NHS arrives
- To forward the packet to its default router

The choice depends on local policies of the LLG to which S belongs. The third solution is recommended as the default choice because it allows the packet to reach D in any case, without forcing S to wait. Obviously, the resolution process is not performed for each packet transmitted to a given destination because clients have a local cache at their disposal.

When the NHS receives the NHRP_Request message from S, it checks whether an entry containing the ATM address of the next hop toward D is present in its cache. If not, the NHS forwards the same request to another NHS. The request passes from NHS to NHS until one of the following conditions occurs:

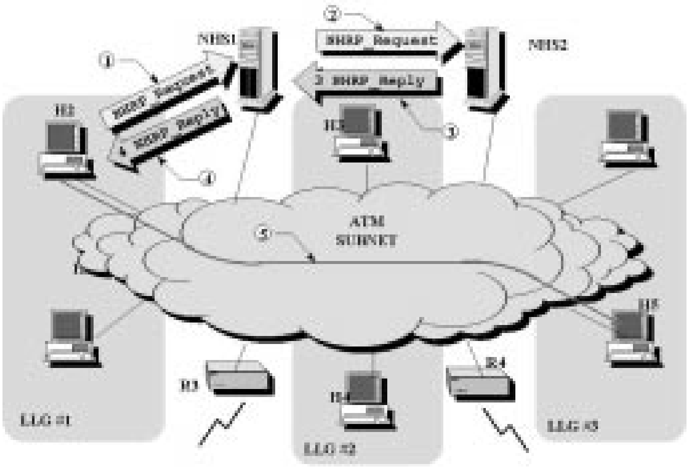
- The request reaches the NHS serving D. This NHS can reply to the request by generating an NHRP_Reply message containing IP and ATM addresses of the next hop toward D. Obviously, if D is not connected to the ATM network, this next hop is the ATM address of the router toward the network where D is located.
- No NHS can resolve the next hop toward D. In this case, the last visited NHS generates a negative NHRP_Reply message.

In both cases, the NHRP_Reply message is sent to S along the same path made by the NHRP_Request so that all NHSs traversed by the reply can insert in their caches the information the reply contains. This capability allows the NHSs to reply to subsequent requests for the same next hop with *nonauthoritative* replies—that is, replies not arriving from the NHS where the client is registered. If a communication attempt based on a nonauthoritative reply fails (probably because some variations on the network occurred), the source station can send a new NHRP_Request requesting an authoritative reply.

An example of the preceding approach is illustrated in Figure 9-12. Host H1 wants to forward a packet to host H5, but H1 doesn't know H5's ATM address. It therefore forwards an NHRP_Request to NHS1, which, nevertheless, doesn't have this information. The request is forwarded to NHS2, which, because the NHS is serving H5, can generate an NHRP_ Reply with the requested ATM address. This reply, returning toward H1, traverses NHS1, allowing it to copy this address in its cache for a future use as a nonauthoritative reply. The reply eventually reaches H1, which then can open a VC with H5.

Moreover, NHRP allows the association of the ATM address of a next hop with an entire IP subnet. For example, if router X is the next hop between station S and station D, this means that X is the egress router to be used to reach all other stations belonging to the same IP subnet of D.

Figure 9-12
Example of ATM address resolution with NHRP



9.3 Alternative Approaches

The approaches described in the preceding sections are based on the principle that the interaction between IPv6 and the underlying ATM network is implemented by using ATM standard signaling primitives—that is, first of all the UNI 3.0/3.1[2]. Some manufacturers, following the IETF proposals for CSRs (*Cell Switching Routers*)[13], decided not to follow this approach and to create alternative signaling protocols that allow more direct interaction between switches and routers. These approaches use only the physical part of the UNI specification but completely avoid signaling procedures. Moreover, they don't use the P-NNI. The control of the network and of the routing remains with routers that use classic IP protocols such as OSPF and BGP for this purpose.

9.3.1 IP Switching

With the term *IP switching,* we usually refer to an approach introduced by Ipsilon Networks (**www.ipsilon.com**)[14] based on two key principles:

- IP routing functions can be added to an ATM switch if an external router is allowed to directly control the ATM switch.
- IP packets can be considered as belonging to flows—that is, to have some characteristics in common. This is particularly true for IPv6 packets having the Flow Label inside them (see Section 3.1.3).

By combining these two ideas, the Ipsilon approach proposes to route IP packets by using routers in a hop-by-hop method, or to create ATM VCs dedicated to them, according to traffic characteristics of flows. For example, packets containing queries and DNS replies benefit from hop-by-hop routing implemented through routers because a DNS flow is short and creating a dedicated VC would have an average cost that is too high, although creating a dedicated VC on ATM switches for routing packets generated by a file transfer is undoubtedly useful.

In general, the traffic can be classified according to two types*: flow-oriented* and *short-lived* (see Table 9-1). For packets belonging to the first type, allocating a dedicated VC on ATM switches is convenient; for those belonging to the second type, allowing hop-by-hop routing through a router is convenient.

The IP switching architecture can be better understood by analyzing Figure 9-13. It consists of ATM switches that are always coupled with an IP router and of *IP gateways* that allow the connection of traditional LANs. IP routers control the routing of IP packets using common routing protocols, such as OSPF and BGP, to compute routing tables. Routers provide for directly routing the short-lived traffic, whereas they order switches to create dedicated VCs for the flow-oriented traffic (for this reason, they are also called *switch controllers*).

The interaction between the different elements of the architecture is provided by two protocols: the GSMP and the IFMP.

The GSMP (*General Switch Management Protocol*), which is described by RFC 1987[15], is used by the router to control the switch. In particular, the router can configure the lookup tables of the switch through the GSMP and therefore control the routing of ATM cells. The IFMP (*Ipsilon Flow Management Protocol*), described in RFC 1953[16], is associated with each link and is used by the destination to communicate to the source the VPI/VCI of the VC on which the IP flow must be forwarded. Note that the determination of the VPI/VCI is always made by the receiver and that,

Table 9-1

Types of IP traffic

Flow-Oriented Traffic	Short-Lived Traffic
File Transfer (FTP)	Names Resolution (DNS)
File Sharing (NFS)	Electronic Mail (SMTP)
Web Access (HTTP)	Network Timing Protocol (NTP)
Virtual Terminal (TELNET)	Post Office Protocol (POP)
Multimedia Voice/Video	Network Management (SNMP)

when a flow is not classified, IP packets are forwarded on the default VC (VPI = 0 e VCI = 15), which, at switch level, is always routed toward the router.

Figure 9-14 shows the architecture of an IP switch—that is, the coupling of an ATM switch and a router (called *IP switch controller*) with the additional modules for the management of IFMP and GSMP protocols and for flow classification.

The short-lived traffic is routed on the default VC; it is conveyed by the ATM switch to the switch controller that, operating like a router, determines the next hop by consulting its IP routing tables, computed by protocols such as OSPF and BGP.

A different approach should be followed for the flow-oriented traffic. It is initially routed on the default VC, but flow-classifier modules that are

Figure 9-13
IP switching architecture

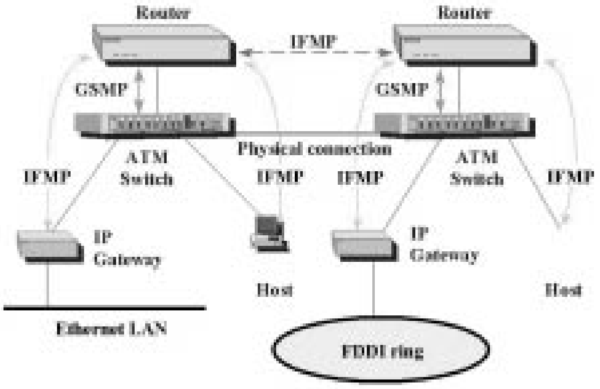
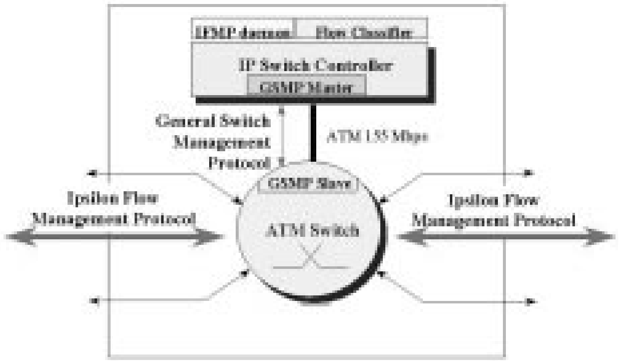


Figure 9-14
IP switch architecture


IFMP daemon
Flow Classifier
IP Switch Controller
GSMP Master
General Switch Management Protocol
ATM 155 Mbps
GSMP Slave
Ipsilon Flow Management Protocol
ATM Switch
Ipsilon Flow Management Protocol

present both on switch controllers and on stations recognize the flow-oriented nature of this traffic and request the creation of a dedicated VC. This VC is created with a series of steps that can be better understood by analyzing the example shown in Figure 9-15.

At the beginning, in phase (1), the traffic is routed on the default VC through the switch controller that rebuilds IP packets starting from ATM cells, consults routing tables, segments packets again, and forwards them to the destination always using the default VC.

When the flow-classifier module of the switch controller recognizes flow-oriented traffic, it requests the switch, through the GSMP protocol, to create a new VC; then it signals to the upstream node through the IFMP protocol to use it (2). The upstream node begins to forward IP packets on the new VC (3), but packets continue to reach the switch controller. Also, the downstream node recognizes the flow-oriented nature of the traffic and requests the switch controller to use a new VC (4). The switch controller begins to use the new VC (5). Eventually, the switch controller realizes that the two dedicated VCs can be interconnected at the switch level; therefore, it programs the switch through the GSMP to directly route cells arriving on the VPI/VCI = 0/X on the VPI/VCI = 0/Y (6). At this point, the cut-through routing is implemented.

In IPv6, the task to classify flows is particularly easy because of the Flow Label field present on IPv6 packets. In fact, the source station itself can indicate whether the traffic is short-lived (Flow Label = 0) or flow-oriented (Flow Label ≠ 0).

9.3.2 Tag Switching

Cisco Systems (**www.cisco.com**) proposes an alternative to IP switching with its technique called *tag switching*. Tag switching is designed to sim-

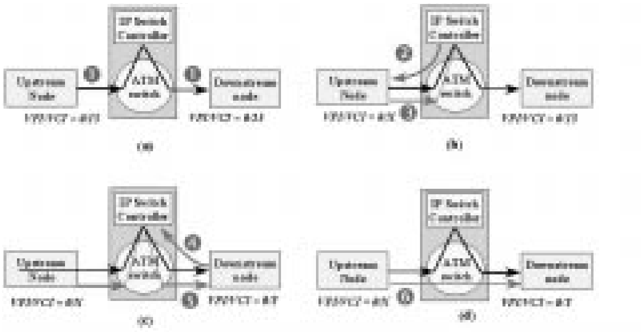


Figure 9-15 Example of the creation of a dedicated VC

plify and to speed routing operations also on non-ATM networks through the subdivision of routing and control functions[17].

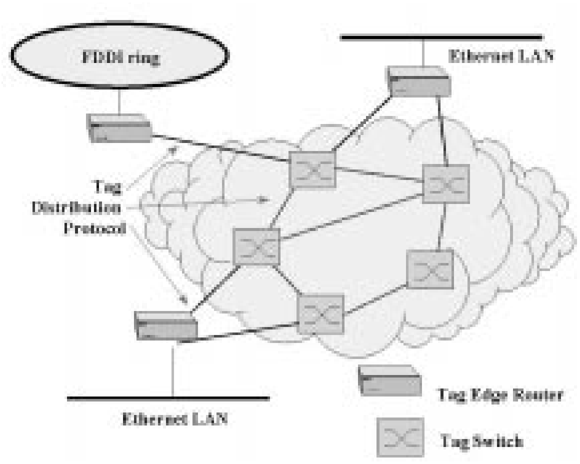
The basic idea is to insert in each packet transmitted on the network an identification, called a *tag*, by which *tag switches* (internetworking devices located between the source and the destination) can implement fast routing (see Figure 9-16). The information contained in tags and that maintained by each tag switch is used to implement the routing; the control, on the other side, is the component of the protocol that is responsible for tables updating within tag switches, and it uses, for this purpose, the TDP (*Tag Distribution Protocol*)[18].

The routing adopted in the tag switching is mainly based on the *label swapping* paradigm. When a packet labeled with a determinate tag is received by a tag switch, this switch uses the tag to examine its TIB (*Tag Information Base*). The TIB is a table in which each entry is formed by an entry tag field and by one or more fields to be used for routing the egress packet. These fields can contain, for example, the tag to be placed on the egress packet, the interface of the switch on which the packet should be transmitted, or further information useful to the layer 2 protocol (for example, the MAC address of the following node).

This routing procedure is extremely simple, and it can be implemented in hardware. Moreover, it is suitable for the management of the multicast at IP level because the same entry tag can be associated with many entries in the TIB.

The main difference between tag switching and IP switching is that in IP switching the presence of IP packets activates the creation of ATM VCs, whereas in tag switching TIBs are created by the existence of an IP

Figure 9-16
Example of network with tag switching



rate independently from the presence of traffic, and therefore all the traffic is treated the same way by the tag switching.

The three possibilities for creating and managing TIBs starting from routing tables are as follow:

- Downstream allocation
- Downstream on-demand allocation
- Upstream allocation

In all three cases, each switch allocates tags by creating the corresponding entries in its TIB for each destination (IP prefix) present within its routing table (*FIB,* or Forwarding Information Base) and creates a connection between FIB and TIB. This connection also allows the association of tags to packets that were originally lacking them.

In the *downstream* allocation scheme, tags are generated and associated with an IP prefix by the node that, on a given link, is located downstream —that is, by the node receiving the traffic. The *downstream on-demand* allocation works in a similar way, but the upstream node requests the downstream node to allocate a tag for a specific IP prefix. In the *upstream* allocation, each upstream node directly allocates tags for each IP prefix known in its FIB.

In all three cases, after an association between a tag and a prefix is created, it is transmitted to the node at the other end of the link.

The mechanism for the diffusion of information for the updating of TIBs can either exploit packets commonly exchanged for the management of routing protocols at the network level (for example, *piggybacking* on BGP) or use the TDP protocol.

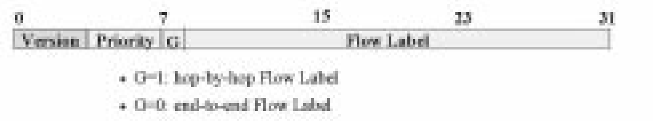
The tag can be transported in a packet in the following three ways, and the choice of the most suitable way depends on the network architecture in which the tag switching is inserted:

- In a proper header between the layer 2 envelope and the layer 3 envelope
- As part of the header of the layer 2 envelope (ATM)
- As part of the header of the layer 3 envelope (IPv6)

In particular, in the IPv6 case, Cisco Systems proposes to transport the tag inside the Flow Label field[19], by partly modifying its meaning, as shown in Figure 9-17.

This proposal introduces a bit G, which discriminates between the original semantic of the Flow Label as proposed in IPv6 (end-to-end) and the semantic necessary for the tag switching (hop-by-hop).

Figure 9-17
Proposal to modify the Flow Label



Moreover, the tag switching allows each packet to carry many tags, in order to obtain a hierarchical routing. These characteristics can be used, for example, to separate the IGP routing information from the EGP routing information.

We can then see that the tag switching of IPv6 packets can be simply implemented on ATM networks. Both techniques are based on tag switching, and a biunivocal or an identity relationship can be established between the couple VPI/VCI and the tag. Tag allocation is implemented by using the downstream on-demand modality.

To allow an ATM classical switch to work like a tag switch, we need to implement classical routing protocols (such as OSPF and BGP), the FIB, the TIB, the TDP, and control modules of the tag switching itself within the switch.

Problems and protocols associated with tag switching and those associated with the traditional ATM signaling (for example, UNI and P-NNI) are independent. We need to create conditions of coexistence between these two schemes and therefore to define a set of VPIs/VCIs to be used with the tag switching and a separate set to be used with the traditional ATM signaling.

A mechanism similar to IP tunneling has been established to eliminate the disadvantage of crossing classical ATM networks, in which intermediate switches unable to manipulate packets marked with tags exist. In this case, two routers that support the tag switching may be interconnected by a Virtual Path and therefore use the VCI like a tag (VP tunneling).

9.3.3 Other Approaches

The great interest aroused by the approaches described in the preceding subsections, added to the lack of precise standards, also urged other companies to propose solutions in this field. Among them, we must mention the following:

- *Cell Switch Router:* This proposal by Toshiba (**www.toshiba.com**) represents the evolution of the work on CSRs[13] originally carried on in Japan. Like tag switching, this proposal is not limited to ATM, but it can operate on other NBMA networks as well and in general on all connection-oriented networks. Like IP switching, it is based on the classification of IP flows and on the creation of bypass pipes. It uses a signaling protocol called FANP (*Flow Attribute Notification Protocol*)[20].

- ARIS: This proposal by IBM (**`www.ibm.com`**) is not limited to ATM, which can operate on other NBMA networks as well and in general on all connection-oriented networks. It uses a signaling protocol called ARIS (Aggregate Route-based IP Switching)[21], which is based on the concept of egress identifiers. ARIS opens some VCs toward each egress identifier, and because thousands of IP destinations can be mapped on a single egress identifier, ARIS minimizes the number of necessary VCs. Each egress router starts the setup of VCs toward its upstream neighbors and these neighbors toward their upstream neighbors using a technique similar to the Reverse Path Multicast. Each router checks the presence of loops on the VC. The VC toward an egress router assumes the form of a tree.
- *SITA* (*Switching IP Through ATM):* This proposal by Telecom Finland (**`www.tele.fi`**) is for ATM networks with two tag levels. It doesn't need a signaling protocol.

REFERENCES

[1]Uyless Black, *ATM: Foundation for Broadband Networks*, Prentice-Hall, 1995.

[2]ATM Forum, *ATM User-Network Interface Specification*, Prentice-Hall, September 1993.

[3]J. Heinanen, *RFC 1483: Multiprotocol Encapsulation over ATM Adaptation Layer 5*, July 1993.

[4]S. Gai, P.L. Montessoro, P. Nicoletti, *Reti Locali: dal Cablaggio all'Internetworking*, SSGRR (Scuola Superiore G. Reiss Romoli), 1995.

[5]G. Armitage, *RFC 2022: Support for Multicast over UNI 3.0/3.1 based ATM Networks*, November 1996.

[6]G. Armitage, *IPv6 and Neighbor Discovery over ATM*, IETF Internet Draft, June 1996.

[7]P. Schulter, *A Framework for IPv6 over ATM*, Internet Draft, February 1996.

[8]R. Atkinson, D. Haskin, J. Luciani, *IPv6 over NBMA Networks*, IETF

Internet Draft, June 1996.

[9] G. Armitage, *Transient Neighbors for IPv6 over ATM*, Internet Draft, June 1996.

[10] A. Conta, S. Deering, *RFC 1885: Internet Control Message Protocol (ICMPv6)*, December 1995.

[11] M. Laubach, *RFC 1577: Classical IP and ARP over ATM*, January 1994.

[12] J. Luciani, D. Katz, D. Piscitello, B. Cole, *NBMA Next Hop Resolution Protocol (NHRP)*, IETF Internet Draft, July 1996.

[13] H. Esaki, M. Ohta, K. Nagami*, High Speed Datagram Delivery over Internet using ATM Technology*, IEEE TRANS. Communications, Vol. E78-B, No. 8, August 1995.

[14] P. Newman, T. Lyon, G. Minshall, *Flow labelled IP: A connectionless approach to ATM*, Proc. IEEE Infocom, San Francisco, March 1996, pp. 1251-1260.

[15] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, G. Minshall, *RFC 1987: Ipsilon's General Switch Management Protocol Specification Version 1.1*, August 1996.

[16] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, G. Minshall, *RFC 1853: Ipsilon Flow Management Protocol Specification for IPv4 Version 1.0*, May 1996.

[17] Y. Rekhter, et al., *Tag Switching Architecture Overview*, Internet Draft, September 1996.

[18] P. Doolan, et al., *Tag Distribution Protocol*, Internet Draft, September 1996.

[19] F. Baker, et al., *Use of Flow Label for Tag Switching*, Internet Draft, August 1996.

[20] Y. Katsube, K. Nagami, H. Esaki, *Router Architecture Extensions for ATM: Overview*, Internet Draft, November 1996.

[21] R. Woundy, A. Viswanathan, N. Feldman, R. Boivie*, ARIS: Aggregate Route-Based IP Switching*, Internet Draft, November 1996.