APPENDIX A

INTRODUCTION TO ATM NETWORKS

Future applications are expected to require increasingly higher bandwidth and generate a heterogeneous mix of network traffic. Existing networks cannot provide the transport facilities to efficiently support a diversity of traffic with various service requirements. ATM is potentially capable of supporting all classes of traffic (e.g., voice, video, and data) in one transmission and switching fabric technology (as represented by figure A.1). It promises to provide greater integration of capabilities and services, increased and more flexible access to the network, and more efficient and economical service.

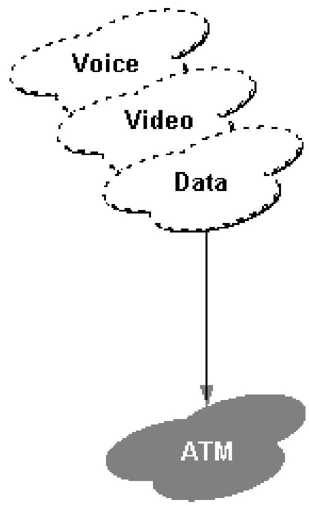

Figure A.1: Multiple networks combine into one

A.1 INTRODUCTION

A.1.1 Basic Principles

ATM stands for *Asynchronous Transfer Mode*. The key innovation here is the word Asynchronous - non-periodic. The telephone networks today use synchronous transfer mode[27]. The clocks at successive nodes (switches) of the networks are synchronized and the information transfer is periodic. Users are assigned slots and every n^{th} slot belongs to the same user. This mode, also known as time-division multiplexing, matches well with circuit switching applications where information arrives periodically. However, for data applications, it is better to be able to transmit asynchronously. This is similar to packet switching and is more suitable for high-speed transfer.

ATM carries all traffics on a stream of fixed-size packets (cells), each comprising 5 bytes of header information and 48 bytes of information field (payload). The reason for choosing a fixed-size packet is to ensure that the switching and multiplexing function could be carried out quickly and easily and to reduce the variance of delay to make the networks suitable for integrated traffic consisting of voice, video and data [10,28].

ATM is a *connection-oriented* technology in the sense that before two systems on the network can communicate, they should inform all intermediate switches about their service requirements and traffic parameters. This is similar to the telephone networks where a fixed path is set up from the calling party to the receiving party. In ATM networks, each connection is called a *virtual circuit* or *virtual channel* (VC), because it also allows the capacity of each link to be shared by connections using that link on a demand basis rather than by fixed allocations. The connections allow the network to guarantee the quality of service (QoS) by limiting the number of VCs. Typically, a user declares key service requirements at the time of connection setup, declares the traffic parameters and may agree to control these parameters dynamically as demanded by the network.

ATM networks use VC Ids (VCIs) to identify cells of various connections. VCIs are assigned locally by the switch. A connection has different VCI at each switch. The same VCI can be used for different connections at different switches. Thus, the size of VCI field does not limit the number of VCs that can be set up. This makes ATM networks scalable in terms of number of connections. In TCP/IP networks, the number of end systems is limited by the size of the address fields making them non-scalable in terms of number of connections [27].

A.1.2 The ATM Forum

With the objective of accelerating the convergence of standards and industry cooperation, an international consortium called the ATM Forum was found to ensure interoperability between public and private ATM implementations and to promote the use of ATM products and services. Although it is not a standards body, the ATM Forum works closely with standards organizations such as the International Telecommunications Union (ITU) and Internet Engineering Task Force (IETF) in developing the definitions for the ATM standards. This international consortium has grown from fewer than ten members in 1991 to over 700 members currently, consisting of public and private network equipment vendors and service providers, software companies, as well as government organizations, national research laboratories, and universities.

A.2 ATM PROTOCOL REFERENCE MODEL

The ATM protocol reference model is based on standards developed by the ITU. Communication from higher layers is adapted to the lower ATM defined layers, which in turn pass the information onto the physical layer for transmission over a selected physical medium. The protocol reference model is divided into three layers: the ATM adaptation layer (AAL), the ATM layer, and the physical layer (figure A.2 [28]).

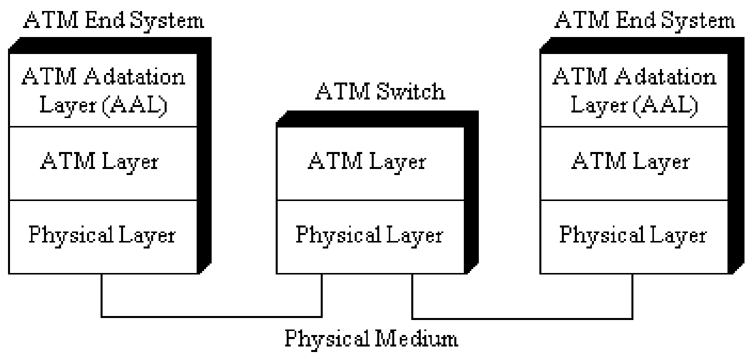

Figure A.2: ATM Protocol Structure

A.2.1 The ATM Adaptation Layer

The ATM Adaptation Layer (AAL) interfaces the higher layer protocols to the ATM Layer. It relays ATM cells both from the upper layers to the ATM Layer and vice versa. When relaying information received from the higher layers to the ATM Layer, the AAL *segments* the data into ATM cells. When relaying information received from the ATM Layer to the higher layers, the AAL must take the cells and *reassemble* the payloads into a format which could be understandable by the higher layers. This is called *Segmentation and Reassembly* (SAR). Four types of AALs had been proposed. Each type supports a different type of traffic or service expected to be used on ATM networks. The service classes and the corresponding types of AALs are as follows:

- *Class A - Constant Bit Rate (CBR) service*: AAL1 supports a connection-oriented service in which the bit rate is constant. Examples of this service include 64 Kbit/sec voice, fixed-rate uncompressed video and leased lines for private data networks.
- *Class B - Variable Bit Rate (VBR) service*: AAL2 supports a connection-oriented service in which the bit rate is variable but requires a bounded delay for delivery.

Examples of this service include compressed packetized voice or video. The requirement on bounded delay for delivery is necessary for the receiver to reconstruct the original uncompressed voice or video.

- *Class C - Connection-oriented data service*: Examples of this service include connection-oriented file transfer and in general, data network applications where a connection is set up before data is transferred. This service has variable bit rate and does not require bounded delay for delivery. The ITU originally recommended two types of AAL protocols to support this class, but merged them into a single type, called AAL3/4. Because of the high complexity of AAL3/4 protocols, the AAL5 protocol has been proposed and is often used to support this class of service.
- *Class D - Connectionless data service*: Examples of this service include datagram traffic and in general, data network applications where no connection is set up before data is transferred. Either AAL3/4 or AAL5 can be used to support this class.

Although each AAL is optimized for a specific type of traffic, there is no stipulation in the standards that AALs designed for one class of traffic cannot be used for another. In fact, many vendors of ATM equipment currently manufacture products that use AAL5 to support all the above classes of traffic, and most activities of the ATM Forum have focused on AAL5. The AAL is also important in the internetworking of different networks and services.

A.2.2 The ATM Layer

The ATM layer provides an interface between the AAL and the physical layer. This layer is responsible for relaying cells from the AAL to the physical layer for transmission and from the physical layer to the AAL for use at the end systems. When it is inside an end system, the ATM layer receives a stream of cells from the physical layer and transmits either cells with new data or empty cells if there is no data to send. When it is inside a switch, the ATM layer determines where the incoming cells should be forwarded to, resets the corresponding connection identifiers and forwards the cells to the next link. Moreover, it buffers incoming and outgoing cells, and handles various traffic management functions such as cell loss priority marking, congestion indication, and generic flow control access. It also monitors the transmission rate and conformance to the service contract (traffic policing).

A.2.3 The Physical Layer

The physical layer defines the bit timing and other characteristics for encoding and decoding the data into suitable electrical/optical waveforms for transmission and reception on the specific physical media used. In addition, it also provides cell delineation function, header error check (HEC) generation and processing, performance monitoring, and payload rate matching of the different transport formats used at this layer.

The Synchronous Optical Network (SONET), a synchronous transmission structure, is often used for framing and synchronization at the physical layer. In addition to the optical media and line rates defined for SONET, the ATM Forum has proposed a variety of physical layer standards, such as ATM over twisted-pair wire. This will accelerate the acceptance of ATM as a desktop connection technology since existing cabling plants can be retained and the cost per connection will be reduced.

A.3 ATM NETWORK OPERATION

An ATM network consists of a set of ATM switches interconnected by point-to-point ATM links or interfaces. ATM switches support two kinds of interfaces: *user-network interfaces* (UNI) and *network-node interfaces* (NNI). UNI connect ATM end-systems (hosts, routers, and so on) to an ATM switch, while an NNI may be imprecisely defined as an interface connecting two ATM switches together; slightly different cell formats are defined across UNI and NNI. More precisely, however, an NNI is any physical or logical link across which two ATM switches exchange the NNI protocol (figure A.3).

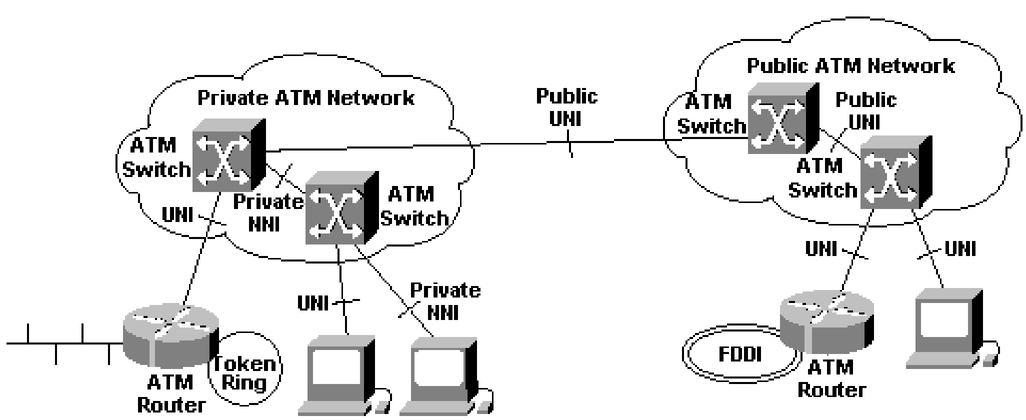

Figure A.3: ATM Network Interfaces

As noted above, ATM networks are fundamentally connection-oriented. This means that a virtual circuit needs to be set up across the ATM network prior to any data transfer. ATM circuits are of two types: *virtual paths*, identified by virtual path identifiers (VPI); and *virtual channels*, identified by the combination of a VPI and a virtual channel identifier (VCI). A virtual path is a bundle of virtual channels, all of which are switched transparently across the ATM network on the basis of the common VPI. All VCI and VPI, however, have only local significance across a particular link, and are remapped, as appropriate, at each switch.

The basic operation of an ATM switch is very simple and could be explained by figures A.4 and A.5. To receive a cell across a link on a known VCI or VPI value; to look up the connection value in a local translation table to determine the outgoing port (or ports) of the connection and the new VPI/VCI value of the connection on that link; and then to retransmit the cell on that outgoing link with the appropriate connection identifiers.

Input		Output	
Port	VPI/VCI	Port	VPI/VCI
1	29	2	45
2	45	1	29
1	64	3	29
3	29	1	64

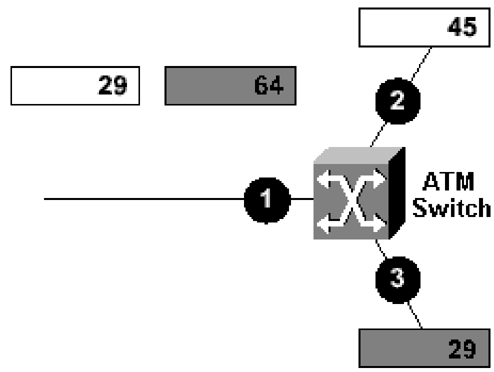

Figure A.4: ATM Switch Operation

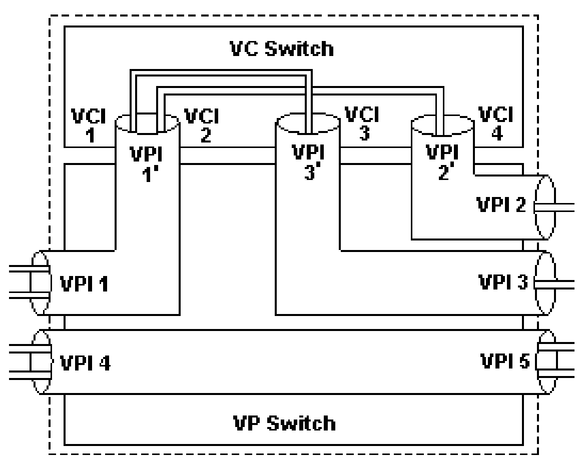

Figure A.5: Virtual Circuit and Virtual Path Switching

The switch operation is so simple because external mechanisms set up the local translation tables prior to the transmittal of any data. The manner in which these tables are set up determine the two fundamental types of ATM connections:

• *Permanent Virtual Connections (PVC)*: A PVC is a connection set up by some external mechanism, typically network management, in which a set of switches between an ATM source and destination ATM system are programmed with the appropriate VPI/VCI values.
• *Switched Virtual Connections (SVC)*: A SVC is a connection that is set up automatically through a signaling protocol. SVCs do not require the manual interaction needed to set up PVCs and, as such, are likely to be much more widely used. All higher layer protocols operating over ATM primarily use SVCs.

ATM signaling is initiated by an ATM end-system that desires to set up a connection through an ATM network; signaling packets are sent on a well known virtual channel, VPI=0, VCI=5. The signaling is routed through the network, from switch to switch as shown in figure A.6, setting up the connection identifiers as it goes, until it reaches the destination end system. The latter can either accept and confirm the connection request, or can reject it, clearing the connection. Note that because the connection is set up along the path of the connection request, the data also flows along this same path.

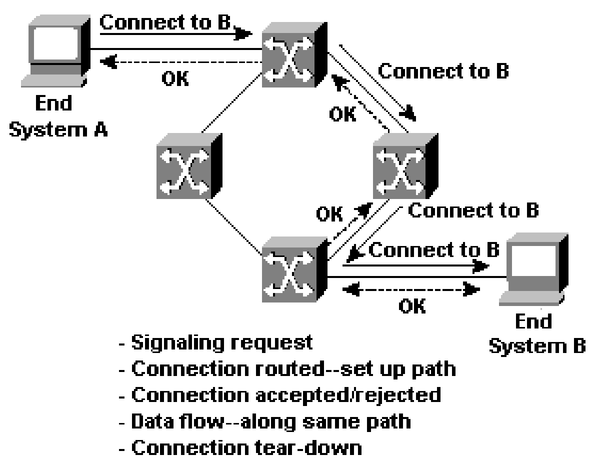

Figure A.6: Connection Setup through ATM Signaling (SVC)

There are two fundamental types of ATM connections (see figure A.7):
• *Point-to-point connection*s, which connect two ATM end-systems. Such connections can be uni-directional or bi-directional.
• *Point-to-multipoint connection*s, which connects a single source end-system (known as the root node) to multiple destination end-systems (known as leaves). Cell replication is done within the network by the ATM switches at which the connection splits into two or more branches. Such connections are uni-directional, permitting the root to transmit to the leaves,

but not the leaves to transmit to the root, or to each other, on the same connection. The reason why such connections are only uni-directional are described below.

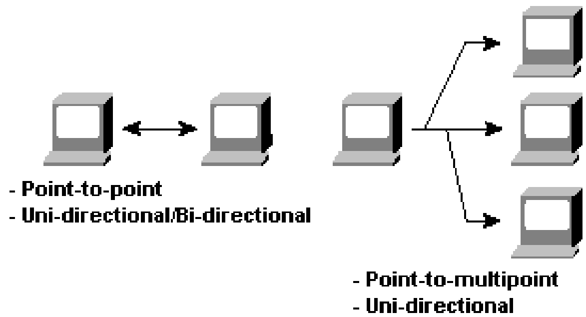

Figure A.7: Types of ATM Connections

What is notably missing from these types of ATM connections is an analog to the multicasting or broadcasting capability common in many shared medium LAN technologies such as Ethernet or Token Ring. In such technologies, multicasting allows multiple end systems to both receive data from other multiple systems, and to transmit data to these multiple systems. Such capabilities are easy to implement in shared media technologies such as LANs, where all nodes on a single LAN segment must necessarily process all packets sent on that segment. The obvious analog in ATM to a multicast LAN group would be a (bi-directional) multipoint-to-multipoint connection. Unfortunately, this obvious solution cannot be implemented when using AAL5, the most common ATM Adaptation Layer (AAL) used to transmit data across ATM networks.

Unlike AAL 3/4, with its Message Identifier (MID) field, AAL 5 does not have any provision within its cell format for the interleaving of cells from different AAL5 packets on a single connection. This means that all AAL5 packets sent to a particular destination across a particular connection must be received in sequence, with no inter-leaving between the cells of different packets on the same connection, or the destination reassembly process would not be able to reconstruct the packets. This is why ATM AAL 5 point-to-multipoint connections can only be unidirectional, thus if a leaf node has to transmit an AAL 5 packet onto the connection, it would be received by both the root node and all other leaf nodes. However, at these nodes, the packet sent by the leaf could well be interleaved with packets sent by the root, and possibly other leaf nodes; this would preclude the reassembly of any of the interleaved packets. Clearly, this is not acceptable.

Despite this problem, ATM does require some forms of multicast capability, since most existing protocols, being developed initially for LAN technologies, rely upon the existence of a low-level multicast/broadcast facility. Three methods have been proposed for solving this problem [29]:

- **VP-Multicasting:** In this mechanism, a multipoint-to-multipoint VP links all nodes in the multicast group, and each node is given a unique VCI value within the VP. Interleaved packets can hence be identified by the unique VCI value of the source. Unfortunately, this mechanism requires a protocol to uniquely allocate VCI values to nodes; such a mechanism does not currently exist. It is also not clear whether current segmentation and reassembly (SAR) devices could easily support such a mode of operation.

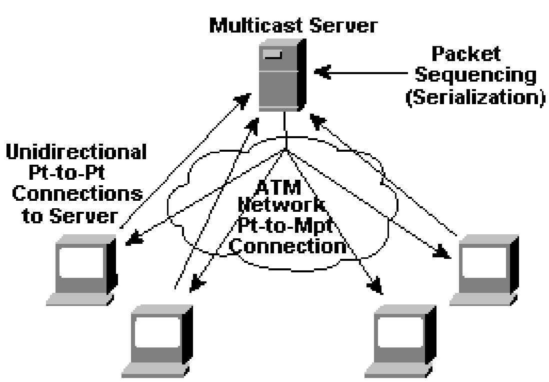

Figure A.8: Multicast Server Operation

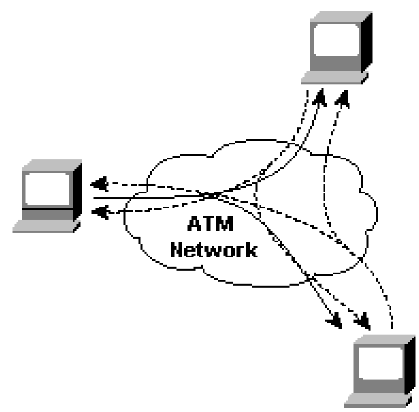

Figure A.9: Overlaid P-to-MP Connections

- **Multicast Server:** In this mechanism, all nodes wishing to transmit onto a multicast group set up a point-to-point connection with an external device known as a multicast server (perhaps better described as a resequencer or serializer). The multicast server, in turn, is connected to all nodes wishing to receive the multicast packets through a point-to-multipoint connection as shown in figure A.8. The multicast server receives packets across the point-to-point connections, then retransmits them across the point-to-multipoint connection but only after ensuring that the packets are serialized (that is, one packet is fully transmitted prior to the next being sent). In this way, cell interleaving is precluded.

- **Overlaid Point-to-Multipoint Connections:** In this mechanism, all nodes in the multicast group establish a point-to-multipoint connection with each other node in the group, and, in turn, becomes a leaf in the equivalent connections of all other nodes. Hence, all nodes can both transmit to and receive from all other nodes. This is shown in figure A.9.

The last mechanism requires each node to maintain N connections for each group, where N is the total number of transmitting nodes within the group, while the multicast server mechanism requires only two connections. This mechanism also requires a registration process for telling nodes that join a group what the other nodes in the group are, so that it can form its own point-to-multipoint connection. The other nodes also need to know about the new node so they can add the new node to their own point-to-multipoint connections. The multicast server mechanism is more scalable in terms of connection resources, but has the problem of requiring a centralized resequencer, which is both a potential bottleneck and a single point of failure.

In short, there is, as yet, no ideal solution within ATM for multicast. Higher layer protocols within ATM networks use both the latter two solutions for multicast. This is one example of why internetworking existing protocols with ATM is so complex. Most current protocols, particularly those developed for LANs, implicitly assume a network infrastructure very similar to existing LAN technologies—that is, a shared medium, connectionless technology with implicit broadcast mechanisms. As noted above, ATM violates all of these assumptions. In the next section the mechanisms used to work around these problems will be discussed.

A.4 LAN EMULATION (LANE)

Before ATM fulfills its promise as a cost-effective technology for supporting future broadband multimedia services, the huge legacy of existing LAN applications needs to be readily migrated to the ATM environment. In other words, ATM needs to provide adequate interworking with legacy LANs.

Various approaches have been proposed to support existing LAN applications in ATM networks. One approach is to consider ATM as a new link layer and to modify the existing network layer protocols to this new technology. An example of this approach is the current work taken by the IETF and ATM Forum on specifying mechanisms for running IP and other network layer protocols over ATM.

Another approach is the provision of an ATM protocol to emulate existing LAN services, allowing network layer protocols to operate as if they are still connected to a conventional

LAN. The LAN emulation (LANE) specification defines how an ATM network can emulate a sufficient set of the medium access control (MAC) services of existing LAN technology (e.g., Ethernet and Token Ring), so that higher layer protocols can be used without modification. Unlike the previous approach, which requires ATM users to change their network operating software, such a LAN emulation service can provide a huge cost benefit. The drawback of this approach, however, is that it also prevents higher layer applications from accessing ATM's unique services.

LAN emulation service would be implemented as device drivers below the network layer in ATM-to-legacy LAN bridges and ATM end systems. In an ATM end system adapter, LAN emulation device drivers would interface with widely accepted driver specifications, such as Network Driver Interface Specification (NDIS) and Open Data link Interface (ODI) used by TCP/IP and IPX.

An important issue related to the LAN emulation service is bandwidth management. Since existing LANs mainly support “best-effort” service, the LAN emulation service also needs to provide a similar capability to its users. This capability is currently supported by the “available bit rate” (ABR) service. ABR service is important not only for LAN emulation, but also for applications involving bursty data traffic where the bandwidth requirements cannot be predicted. This will be discussed in more details in the chapter 2.

A.4.1 LAN Emulation for Connectionless Service

A major difference between existing LANs and ATM networks is that LANs are connectionless, whereas ATM natively supports only connection-oriented services. Thus, an important function for a LANE service is to be able to support a connectionless service over ATM.

Existing LAN services are typically based on a shared medium. When a source system sends a data frame to a destination, it simply adds the destination address to the frame and broadcasts it to the network. A receiving system will then accept the frame when the destination address included in the frame matches its own address. In a network with multiple LAN segments, bridges and routers are used to handle the forwarding of the frame to the segment to which the destination system is attached.

In a connection-oriented network such as ATM, however, a source system needs to first set up a connection to the destination before it can transfer data frames. This requires the source system to exchange control information with the network using a signaling protocol.

It should be noted that LANE is not designed to provide translational bridging between legacy LANs. A bridge or router is still required to internetwork the different legacy LANs. It will not perform all MAC-layer functions either. For example, there will be no need to perform collision detection on an ATM network.

A.4.2 LANE Architecture

LANE consists of three major components:

- LANE client (LEC) - it is generally located in ATM end systems and serves as a proxy for LAN systems,
- LANE server (LES) - its main function is the support of the LAN Emulation address resolution protocol (LE-ARP), needed by a source LEC to determine the ATM address of the target LEC responsible for a certain destination MAC address.
- Broadcast/Unknown Server (BUS) - its task is to forward all multicast traffic to all attached LECs.

The message transfer algorithm for a frame with an individual destination MAC address is described as follow [28]:

1. An LEC maintains an address resolution table that implements a mapping between destination MAC addresses and "data direct" VC. Upon receiving a message from a higher layer, an LEC will first attempt to send the message directly to the destination LEC over a data direct VC.

2. If the destination MAC address is not in the address resolution table, the source LEC sends a LE-ARP request to the LES for the ATM address of the destination.

3. After receiving the LE-ARP reply from the LES containing the target ATM address, the LEC updates its address resolution table. Then it establishes a direct VC to that ATM address for transmitting the message. Alternatively, the LEC could send in parallel to the LE-ARP request the initial message with the unknown destination MAC address to the BUS, which will then broadcast the message to all attached LECs.

In the case of a frame with single destination MAC address, the above option of sending messages via the BUS is provided to reduce the message transfer delay caused by the LE-ARP and connection setup procedures and to facilitate the internetworking with stations attached to legacy LANs. In the case of a frame with a multicast MAC address, the source LEC simply sends it to the BUS for broadcasting to all attached stations.

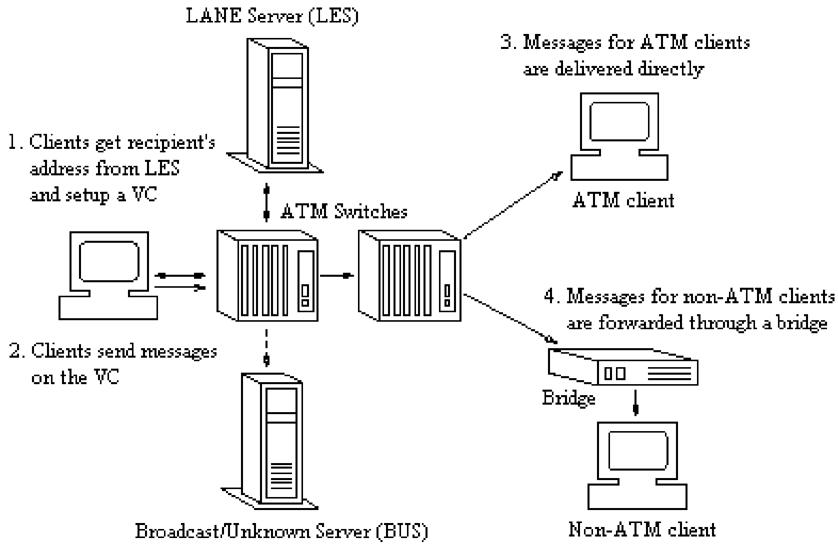

Figure A.10: LANE Architecture

The LAN emulation architecture is illustrated in figure A.10 [28]. An in-depth overview of the main architectural issues related to the provision of LAN emulation service over ATM can be found in [29].